At the Top of the Hypercube – Better Size-Time Tradeoffs for Hash-Based Signatures

Dmitry Khovratovich [1] Mikhail Kudinov*[2] Benedikt Wagner [1]

June 4, 2025

[1] Ethereum Foundation
{dmitry.khovratovich,benedikt.wagner}@ethereum.org
[2] Eindhoven University of Technology
mishel.kudinov@gmail.com

Abstract

Hash-based signatures have been studied for decades and have recently gained renewed attention due to their post-quantum security. At the core of the most prominent hash-based signature schemes, XMSS and SPHINCS$^+$, lies a one-time signature scheme based on hash chains due to Winternitz. In this scheme, messages are encoded into vectors of positions (i.e., vertices in a hypercube) in the hash chains, and the signature contains the respective chain elements. The encoding process is crucial for the efficiency and security of this construction. In particular, it determines a tradeoff between signature size and computational costs. Researchers have been trying to improve this size-time tradeoff curve for decades, but all improvements have been arguably marginal.

In this work, we revisit the encoding process with the goal of minimizing verification costs and signature sizes. As our first result, we present a novel lower bound for the verification cost given a fixed signature size. Our lower bound is the first to directly apply to general encodings including randomized, non-uniform, and non-injective ones.

Then, we present new encodings and prove their security. Inspired by our lower bound, these encodings follow a counterintuitive approach: we map messages non-uniformly into the top layers of a much bigger hypercube than needed but the encoding itself has (hard to find) collisions. With this, we get a 20% to 40% improvement in the verification cost of the signature while keeping the same security level and the same size. Our constructions can be directly plugged into any signature scheme based on hash chains, which includes SPHINCS$^+$ and XMSS.

Keywords: Hash-based signatures, One-time signatures, Post-quantum, Winternitz, incomparable encodings, lower bounds

*Mikhail Kudinov was supported by an NWO VIDI grant (Project No. VI.Vidi.193.066).

Contents

1 Introduction

Hash-based signatures are celebrated for their conceptual simplicity, plausible resistance to quantum attacks, and reliance on conservative assumptions. The key techniques in this area date back to works by Lamport [Lam79], Merkle [Mer79], and Goldreich [Gol87], which serve as the basis for modern constructions, such as SPHINCS$^+$ [HBD+22] and XMSS [BDH11, HBG+18].

Signatures via Hash Chains. The core of hash-based signatures like SPHINCS$^+$ and XMSS is a one-time signature scheme constructed using hash chains. In this scheme, the (one-time) secret key consists of v random strings $\mathsf{sk}_1, \ldots, \mathsf{sk}_v$, which are transformed into the (one-time) public key $\mathsf{pk} = (\mathsf{pk}_1, \ldots, \mathsf{pk}_v)$ through hash chains: $\mathsf{pk}_i := \mathsf{H}^{\mathsf{w}-1}(\mathsf{sk}_i)$, where H is a hash function. Each hash chain has a length of w elements. The signature comprises one element per chain, $\sigma_i := \mathsf{H}^{x_i-1}(\mathsf{sk}_i)$, where the positions $x_1, \ldots, x_v \in [\mathsf{w}]$ are derived from the message. To verify the signature, the verifier checks $\mathsf{H}^{\mathsf{w}-x_i}(\sigma_i) = \mathsf{pk}_i$ for every i.

Clearly, a tradeoff exists between signature size and computational work: reducing v decreases the signature size but necessitates increasing[1] w, thereby raising the total computational cost of signing and verification. While minimizing signature size has traditionally been the primary focus of research, computational efficiency – particularly in verification – has recently gained attention in modern applications [KCLM22, DKKW25]. For example, in the context of succinct arguments that prove knowledge of valid signatures, the prover's efficiency is directly tied to the number of hash invocations required during signature verification.

Incomparable Encodings. A crucial component of one-time signature schemes is the *encoding function* $f\colon \mathcal{M} \to [\mathsf{w}]^v$, which maps messages to positions $x_1, \ldots, x_v$ in the hypercube $[\mathsf{w}]^v$. The parameters v, w, and the choice of f are pivotal, as they directly dictate the scheme's efficiency (signature size, computational cost) and security. At a minimum, the encoding must be *incomparable*: for any two distinct messages $m \neq m'$, their encodings $x = f(m)$ and $x' = f(m')$ must not satisfy $x_i \leq x'_i$ for all $i \in [v]$. Otherwise, an adversary could easily forge a signature for m' given a signature for m. Two common approaches to achieve incomparability are: (1) adding a checksum [Mer79, Mer88], or (2) ensuring all encodings have the same sum $\sum_i x_i$. Equivalently, the latter approach makes all encodings be on the same *layer* of the hypercube. This approach dates back to ideas by Bos and Chaum [BC93] and Vaudenay [Vau93] in the early 90s, and has since been revisited and re-invented [CYK16, PZC+21, HKRY23, ZCY23]. Given that the core ideas of these encodings have been known for over three decades, it is natural to ask whether better encodings are possible. This is the question we explore in this work.

Zhang, Cui, and Yu. This question also motivated Zhang, Cui, and Yu to study incomparable encodings [ZCY23]. Their work focuses on the *rate* $|\mathcal{M}|/|f(\mathcal{M})|$, i.e., the ratio between the size of the message space and the encoding's image. They demonstrate that, among all *deterministic injective encodings*, the constant sum encoding into the largest layer of the hypercube achieves the optimal rate. This result also implies a lower bound on the signature size[2]. However, the constant sum encoding into the largest layer suffers from significant computational verification costs, making it less practical for certain applications. Moreover, the restriction to deterministic injective encodings is unnecessarily limiting. In fact, non-injective encodings can also yield secure schemes, provided they satisfy a suitable form of (target) collision resistance [DKKW25]. As a result, the relationship between verification cost, security, and signature size for general encodings remains poorly understood.

1.1 Our Contribution

In this work, we investigate whether improved encodings can be devised that strike a better balance among signature size, security, and verification efficiency. Specifically, our starting point is the following concrete question:

Fix a signature size and a security level.
How efficient can verification become by optimizing the choice of the encoding function f?

We consider general encoding functions, including those that are randomized, non-injective, or non-uniform.

[1] This tradeoff arises because security imposes a lower bound on $v \log \mathsf{w}$.

[2] This lower bound arises from assuming a minimal message space size, e.g., $|\mathcal{M}| \geq 256$.

Novel Lower Bound. Our first contribution is a novel lower bound on verification cost. Given parameters[3] w, v, and a fixed security level, we consider all encoding functions f mapping into the hypercube $[\mathsf{w}]^v$ and achieving the desired security level. We derive a lower bound on the verification cost for such encodings. Informally, our lower bound demonstrates that the verification cost of any encoding is at least as high as that of a hypothetical[4] encoding $\hat{f}$. This hypothetical encoding maps non-uniformly into the top d_0 layers of the hypercube, with the probability of hitting a fixed point decreasing linearly as one moves to lower layers (which incur higher verification costs). To the best of our knowledge, this is the first lower bound of this kind. In particular, in contrast to Zhang et al. [ZCY23], our lower bound incorporates verification costs and applies to more general classes of encodings, including randomized and non-injective variants.

Better Encodings. Complementing the lower bound, we propose new encodings and prove their security in the random oracle model. These encodings draw inspiration from the structure of the hypothetical encoding $\hat{f}$ used in our lower bound. Somewhat counterintuitively, our results show that using large hypercubes with non-injective encodings can outperform smaller hypercubes with injective encodings in terms of verification efficiency. By applying our proposed encodings and lower bound to concrete parameter sets, we demonstrate significant improvements over classical encodings. Furthermore, the verification costs for our encodings almost match the lower bound. For instance, at a 128-bit security level and assuming 132 chains, classical encodings lead to 66 hash invocations whereas our encodings require around 40 (40% improvement), on average. Assuming 40 chains, classical encodings require 180 hashes. Our encodings improve that to around 135 hashes (25% improvement). Our encodings can be directly plugged into constructions like SPHINCS$^+$ and XMSS, thereby improving their efficiency.

1.2 Extensions and Future Work

We leave several interesting research directions for future work. First of all, it would be natural to translate our results from efficient signature verification to efficient signature generation, i.e., to focus on the problem of minimizing signer hashing. Intuitively, these problems are dual, as for the class of schemes we consider, the fastest signer corresponds to the slowest verifier, and vice versa. We expect that our lower bound can easily be translated to this setting. Similarly, all our encodings would be turned into their duals by applying them to the opposite side of the hypercube.

A second direction might be to devise a lower bound for incomparable encodings only. That is, our bound currently applies to all encodings, but only incomparable ones are of interest. By considering only incomparable encodings, one may strengthen the bound and tighten the gap between the bound and the explicit non-uniform incomparable encodings that we suggest. We are currently unable to come up with such a strengthened bound.

Thirdly, it might be interesting to get some approximation for the lower bound value, which would not involve computing sums of large binomial coefficients. One way to achieve this might be to approximate the layer size through the Central Limit Theorem, so that the encoding cost can be computed as an integral of an exponential function. We leave this mathematical exercise to the reader.

Finally, one should be able to apply our encodings to practical constructions such as XMSS and various versions of SPHINCS$^+$. We note that the parameter selection in those schemes is quite involving (notably, it depends on the hypercube parameters), and extra care must be taken when using our encodings in these settings.

1.3 More on Related Work

In this paper, we study hash-based digital signature schemes. For a comprehensive introduction to this topic, we refer to the book by Boneh and Shoup [BS20]. The work by Bleichenbacher and Maurer [BM94] shows how to generalize signatures based on hash chains to signatures based on directed acyclic graphs. They show under which conditions (similar to incomparability) the resulting signature schemes are secure. We can view our work as studying possible modifications to the Winternitz one-time signature scheme based on hash chains. This scheme has been mentioned first by Merkle [Mer88] and is attributed to Winternitz. The modern version of it is often called WOTS [Hül13]. The state-of-the-art version of WOTS is WOTS-TW, as described in [BHK$^+$19, HK22], which is a part of the standardized hash-based

[3]The parameter v determines the signature size.

[4]The term *hypothetical* indicates that this encoding is not necessarily efficient or would yield a secure scheme.

signature scheme SPHINCS$^+$ [BHK+19], also known as SLH-DSA [HBD+22]. These variants differ mostly in the way the hash function is invoked within the hash chains. In [HKRY23], the authors suggested searching for a message digest that is mapped to chain positions, which sum up to a specific value. This can be viewed as an instantiation of the constant sum encoding mentioned in the introduction, which dates back to Bos and Chaum [BC93] and Vaudenay [Vau93], and has been studied extensively [CYK16, PZC+21, HKRY23, ZCY23]. In [BHRvV21], the authors aimed for faster verification, which is also one of our goals. To this end, the authors propose trying several message digests during signing by iterating a counter and greedily choosing the one that minimizes verification work. The question of an optimal encoding has been studied in [ZCY23]. As we have already explained, the authors investigated only injective encodings and their optimality regarding encoding rate, which is a relation between signature size and message space. Many applications aim to lower verification complexity. As we have mentioned, this was the aim of [BHRvV21]. Recent works [KCLM22, DKKW25] also aimed for minimizing verification hashes to achieve better efficiency when proving knowledge of signatures. Proving knowledge of signatures is relevant for example to construct an aggregation signature scheme [DKKW25].

1.4 Outline

In Section 2, we describe our framework. This includes relevant definitions and shows the connection between signatures based on hash chains and encoding functions mapping into the hypercube. Then, in Section 3, we state and prove our lower bound. It is written for encoding functions and implies a lower bound for signatures based on hash chains via the connections established in Section 2. In Section 4, we present our positive result. Finally, in Section 5 we evaluate our positive result in comparison to our lower bound and earlier encodings.

2 Framework: Encoding into Hypercubes

The central objects that we study in this work are encoding functions that map messages to be signed into the hypercube $[\mathsf{w}]^v$. The resulting string can then be used to compute or verify a signature based on hash chains. This abstraction generalizes Winternitz one-time signatures and has been used by previous works [ZCY23, DKKW25] in different variants. In this section, we precisely define what we mean by the hypercube and encoding functions, and how they can be used in the context of signatures.

2.1 Hypercubes and Encoding Functions

Here, we define the hypercube and adjacent notation that will be relevant to present our results. We start with a more detailed definition of the hypercube.

Definition 1 (Hypercube)**.** Let w and v be integers. The hypercube $[\mathsf{w}]^v$ is the set of all strings of length v over the alphabet $[\mathsf{w}] = \{1, \dots, \mathsf{w}\}$. It induces a directed graph that has vertex set $[\mathsf{w}]^v$ and an edge from x to y if and only if there exists an $i \in [v]$ with $y_i = x_i + 1$ and $x_j = y_j$ for all $j \neq i$.

Definition 2 (Partial Order on the Hypercube)**.** Let w and v be integers and consider the hypercube $[\mathsf{w}]^v$. Consider two vertices $x, x' \in [\mathsf{w}]^v$. We write $x \leq x'$ if and only if there is a directed path from x to x'. In particular, this holds if and only if for every $i \in [v]$, we have $x_i \leq x'_i$.

Looking at the hypercube as a directed graph, it has a single source vertex $(1, \dots, 1)$ and a single sink vertex $(\mathsf{w}, \dots, \mathsf{w})$. We can further partition the other vertices into layers, which are intuitively given by starting a breadth-first search from the source vertex.

Definition 3 (Hypercube Layers)**.** Let w and v be integers and consider the hypercube $[\mathsf{w}]^v$. The layer $\mathcal{L}_d \subseteq [\mathsf{w}]^v$ is the set of vertices at distance d from the sink $(\mathsf{w}, \dots, \mathsf{w})$. Precisely, that is

$$\mathcal{L}_d := \left\{ (x_1, x_2, \dots, x_n) \in [\mathsf{w}]^v \;\middle|\; v\mathsf{w} - \sum_{i=1}^{v} x_i = d \right\}.$$

For $x \in \mathcal{L}_d$, we write $\mathrm{layer}(x) = d$. Note that d can range from 0 to $v(\mathsf{w} - 1)$.

Remark 1. It is easy to see that for any two distinct vertices $x, x' \in \mathcal{L}_d$, we neither have $x \leq x'$ nor $x \geq x'$.

Definition 4 (Sizes of Hypercube Layers)**.** Let w and v be integers and consider the hypercube $[\mathsf{w}]^v$. We denote the size of the dth layer of the hypercube as $\ell_d = |\mathcal{L}_d|$. Note that the size ℓ_d of the dth layer is given by the coefficient of $X^{v\mathsf{w}-d}$ in the polynomial $(X + X^2 + \cdots + X^{\mathsf{w}})^v$. Further, we use the notation $\ell_{[d_1:d_2]} = \sum_{d=d_1}^{d_2} \ell_d$ to denote the joint size of multiple adjacent layers.

With the definition of hypercubes in mind, we can now formally define the concept of an encoding function. Note that previous works [ZCY23, DKKW25] have explicitly required that this function is incomparable, which is necessary for the security of the resulting one-time signature. We decided to define incomparability as an additional feature. This will be useful when we present our lower bounds, as we will first obtain them for any encoding function, and so it will in particular also hold for the incomparable ones that are actually of interest.

Definition 5 (Encoding Function)**.** Let w and v be integers and consider the hypercube $[\mathsf{w}]^v$. An encoding function with domain $\mathcal{M}$ and seed space $\mathcal{R}$ is a function of the form

$$f\colon \mathcal{M} \times \mathcal{R} \to [\mathsf{w}]^v.$$

We also allow f to have the form $f\colon \mathcal{X} \to [\mathsf{w}]^v$ in which case we say f is deterministic. Formally, this is just the special case in which $\mathcal{R}$ has exactly one element. We define the code associated to f as its image

$$\mathcal{C}_f := \{f(m, r) \mid m \in \mathcal{M}, r \in \mathcal{R}\}.$$

Definition 6 (Incomparability)**.** Let w and v be integers and consider the hypercube $[\mathsf{w}]^v$. Let $f\colon \mathcal{M} \times \mathcal{R} \to [\mathsf{w}]^v$ be an encoding function. We say that f is incomparable, if for any distinct $x \neq x' \in \mathcal{C}_f$, we neither have $x \leq x'$ nor $x \geq x'$.

The incomparability definition is concerned with *distinct* codewords x, x'. For such codewords, it will guarantee that an adversary cannot use a signature associated with x to produce a signature associated with x' or vice versa. However, a bad encoding function may map many messages m to the same codeword x, which would also lead to trivial attacks. To rule this out, only injective encodings have been considered in [ZCY23], whereas a security game has been defined in [DKKW25]. We do not want to restrict ourselves to injective encodings. Instead, we follow [DKKW25] by using a target collision-resistance notion for the encoding.

Remark 2 (Units of Time). In the following definition (and others concerned with security), we talk about adversaries $\mathcal{A}$ running in time t. By that, we mean that they run in some default computational model and use t time steps. Our results work for any such model, but we will for ease of notation assume that sampling a random input to the encoding and computing the result of the encoding can be done within one unit of time. For our positive results, i.e., our new encodings, we will assume that t denotes the number of random oracle queries.

Definition 7 (Target Collision Resistance)**.** Let w and v be integers and consider the hypercube $[\mathsf{w}]^v$. Let $f\colon \mathcal{M} \times \mathcal{R} \to [\mathsf{w}]^v$ be an encoding function. Let $\epsilon\colon \mathbb{R} \to \mathbb{R}$ be a function. We say that f is ϵ-secure with respect to target collision resistance, if for any algorithm $\mathcal{A}$ running in time t, we have

$$\Pr\left[f(m, r) = f(m^*, r^*) \wedge m^* \neq m \;\middle|\; \begin{array}{c} (m, St) \leftarrow \mathcal{A}, \\ r \xleftarrow{\$} \mathcal{R},\ (m^*, r^*) \leftarrow \mathcal{A}(St, r) \end{array}\right] \leq \epsilon(t),$$

Remark 3 (Extra Inputs). In [DKKW25], incomparable encoding functions have been used to define a generalized XMSS construction. As the authors have been concerned with proving concrete security for this construction under precise standard model properties of underlying hash functions, they give additional inputs to the encoding function, which we omit. In our work, we aim to understand the fundamental properties of encoding functions, and therefore omit such additional inputs for conceptual simplicity.

2.2 Signatures based on Hash Chains and Encodings

We now recall the well-known method of using hash chains to construct one-time signatures. This is done in the prominent Winternitz one-time signature and its variants, all of which can be abstractly written using an encoding function. We start by recalling the notion of a one-time signature scheme.

Definition 8 (One-Time Signature Scheme)**.** A one-time signature scheme is a tuple of algorithms $\mathsf{SIG} = (\mathsf{Gen}, \mathsf{Sig}, \mathsf{Ver})$ with the following syntax:

- $\mathsf{Gen}(\mathsf{par}) \to (\mathsf{pk}, \mathsf{sk})$ takes as input parameters par and outputs a public key pk and a secret key sk.
- $\mathsf{Sig}(\mathsf{sk}, \mathsf{m}) \to \sigma$ takes as input a secret key sk and a message $\mathsf{m} \in \mathcal{M}$ and outputs a signature σ.
- $\mathsf{Ver}(\mathsf{pk}, \mathsf{m}, \sigma) \to b$ is deterministic, takes as input a public key pk, a message $\mathsf{m} \in \mathcal{M}$, and a signature σ, and outputs a bit $b \in \{0, 1\}$.

Further, SIG should satisfy the following properties in terms of correctness and security:

- *Correctness.* For all $(\mathsf{pk}, \mathsf{sk}) \in \mathsf{Gen}(\mathsf{par})$, and all messages $\mathsf{m} \in \mathcal{M}$ we have

$$\Pr\left[\mathsf{Ver}(\mathsf{pk}, \mathsf{m}, \sigma) = 1 \mid \sigma \leftarrow \mathsf{Sig}(\mathsf{sk}, \mathsf{m})\right] = 1.$$

- *One-Time Security.* For every algorithm $\mathcal{A}$ running in time t, we have

$$\Pr\left[\mathsf{Ver}(\mathsf{pk}, \mathsf{m}^*, \sigma^*) = 1 \wedge \mathsf{m}^* \neq \mathsf{m} \;\middle|\; \begin{array}{c} (\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(\mathsf{par}),\ (\mathsf{m}, St) \leftarrow \mathcal{A}(\mathsf{par}, \mathsf{pk}), \\ \sigma \leftarrow \mathsf{Sig}(\mathsf{sk}, \mathsf{m}),\ (\mathsf{m}^*, \sigma^*) \leftarrow \mathcal{A}(St, \sigma) \end{array}\right] \leq \epsilon(t),$$

  where $\epsilon\colon \mathbb{R} \to \mathbb{R}$ is a function that determines the security level of the scheme. Concretely, in this case we call SIG ϵ-secure.

We present the following construction to recall how to construct one-time signatures from hash chains. For simplicity, and to not distract from our analysis of encoding functions, we present the construction using hash functions modeled as random oracles. We note, however, that modern variants of this rely on so-called *tweakable hash functions*, and we expect all of our results to naturally carry over to that setting as well.

Construction 1 (One-Time Signatures Based on Hash Chains)**.** *Let* $\mathsf{H}\colon \mathcal{X} \to \mathcal{X}$ *be a random oracle. Let* w *and* v *be integers and consider the hypercube* $[\mathsf{w}]^v$*. Let* $f\colon \mathcal{M} \times \mathcal{R} \to [\mathsf{w}]^v$ *be an encoding function. With that, consider the following construction* $\mathsf{SIG} = (\mathsf{Gen}, \mathsf{Sig}, \mathsf{Ver})$*:*

- $\mathsf{Gen}(\mathsf{par}) \to (\mathsf{pk}, \mathsf{sk})$*:*
  1. *For* $i \in [v]$*:* $\mathsf{sk}_i \xleftarrow{\$} \mathcal{X}$*,* $\mathsf{pk}_i := \mathsf{H}^{\mathsf{w}-1}(\mathsf{sk}_i)$
  2. $\mathsf{pk} := (\mathsf{pk}_1, \ldots, \mathsf{pk}_v), \mathsf{sk} := (\mathsf{sk}_1, \ldots, \mathsf{sk}_v)$
- $\mathsf{Sig}(\mathsf{sk}, \mathsf{m}) \to \sigma$ *for* $\mathsf{sk} = (\mathsf{sk}_1, \ldots, \mathsf{sk}_v)$*:*
  - $r \xleftarrow{\$} \mathcal{R}$*,* $x = (x_1, \ldots, x_v) := f(\mathsf{m}, r)$
  - *For* $i \in [v]$*:* $\sigma_i := \mathsf{H}^{x_i - 1}(\mathsf{sk}_i)$
  - $\sigma := (r, \sigma_1, \ldots, \sigma_v)$
- $\mathsf{Ver}(\mathsf{pk}, \mathsf{m}, \sigma) \to b$ *for* $\mathsf{sk} = (\mathsf{sk}_1, \ldots, \mathsf{sk}_v)$ *and* $\sigma = (r, \sigma_1, \ldots, \sigma_v)$*:*
  - $x = (x_1, \ldots, x_v) := f(\mathsf{m}, r)$
  - *For* $i \in [v]$*:* $\mathsf{pk}'_i := \mathsf{H}^{\mathsf{w}-x_i}(\sigma_i)$
  - $b := (\mathsf{pk}_1 = \mathsf{pk}'_1) \wedge \cdots \wedge (\mathsf{pk}_v = \mathsf{pk}'_v)$

Remark 4 (Security). It is known that if the encoding function f is injective and incomparable (cf. Definition 6), and the hash H is *one-way on iterates*, then SIG is ϵ-secure [BS20]. Concretely, this holds if H is modeled as a random oracle for $\epsilon \leq O(\mathsf{w}Q_{\mathsf{H}}/|\mathcal{X}|)$, where Q_{H} denotes the number of queries to H [BS20]. In our case, we will not restrict ourselves to injective encoding functions. Instead, one can see (see e.g., [DKKW25]) that the scheme is ϵ-secure if f is incomparable and ϵ'-secure with respect to target collision resistance (cf. Definition 7) for

$$\epsilon' \leq \epsilon \leq \epsilon' + O(\mathsf{w}Q_{\mathsf{H}}/|\mathcal{X}|).$$

To see the easy direction, note that breaking target collision resistance of the encoding directly translates in to a forgery for the scheme. For the other inequality, we can either turn an attacker against the scheme into one against target collision resistance, or it can be used to break the one-way on iterates property. Hence, we only need to focus on target collision resistance of the encoding function f.

2.3 Cost Function and Constraints

With Construction 1 and Remark 4 in mind, we now define a cost function for encoding functions. We also define a metric for encoding functions and show that it tightly relates to the security level of Construction 1. In this way, we will be able to derive our lower bounds by lower bounding the cost function for any encoding function with a certain metric.

Signature Verification as a Cost Function. We start by presenting the cost function. It associates a cost to any given encoding function. Intuitively, this cost corresponds to the average amount of work that algorithm Ver in Construction 1 has to do. Namely, suppose the encoding $x = f(m, r)$ is in the dth layer of the hypercube, i.e., $x \in \mathcal{L}_d$. Then, by definition $v\mathsf{w} - \sum_{i=1}^{v} x_i = d$. The verifier in Construction 1 has to evaluate H a total amount of $\sum_{i=1}^{v}(\mathsf{w} - x_i) = d$ times. As applications of Construction 1 will additionally hash the public key, our cost function also includes a term $\widetilde{C_v}$. We do not specify this term but note that our results hold for any $\widetilde{C_v}$ which only depends on v and is non-decreasing in v.

Definition 9 (Cost Function)**.** Let w and v be integers and consider the hypercube $[\mathsf{w}]^v$. Let $f\colon \mathcal{M} \times \mathcal{R} \to [\mathsf{w}]^v$ be an encoding function. For any pair $(m, r) \in \mathcal{M} \times \mathcal{R}$, we define its cost with respect to f as

$$\mathsf{cost}(f, m, r) := \widetilde{C_v} + \mathrm{layer}(f(m, r)).$$

The cost of f is defined as

$$\mathsf{cost}(f) := \mathop{\mathbb{E}}_{m,r}[\mathsf{cost}(f, m, r)] = \widetilde{C_v} + \mathop{\mathbb{E}}_{m,r}[\mathrm{layer}(f(m, r))],$$

where the expectation is over $m \xleftarrow{\$} \mathcal{M}$ and $r \xleftarrow{\$} \mathcal{R}$.

Metrics and Security. There are multiple goals one could have in mind when picking an encoding function. For example, one could try to minimize the signature size by lowering the number of chains v, one could try to improve signing times, and more. In our work, we focus on verification cost. With this in mind, one of our goals is to derive lower bounds for $\mathsf{cost}(f)$, which gives us lower bounds on the verification cost of Construction 1. Of course, one could define an encoding function that only maps into the $d = 0$ layer, which results in minimal verification cost $\mathsf{cost}(f) = C_v$. However, such an encoding would certainly not yield a secure scheme in Construction 1. Therefore, to get meaningful lower bounds for $\mathsf{cost}(f)$, we should only consider encoding functions f that can plausibly have the desired security level. As we will see below, a sufficiently low collision metric, as defined next, is a necessary condition to achieve a certain security level.

Definition 10 (Collision Metric of Encoding Functions)**.** Let w and v be integers and consider the hypercube $[\mathsf{w}]^v$. Let $f\colon \mathcal{M} \times \mathcal{R} \to [\mathsf{w}]^v$ be an encoding function. We define its collision metric as

$$\mu_{\ell^2}(f) := \sum_{x \in [\mathsf{w}]^v} \left(\Pr_{m,r}[f(m, r) = x] \right)^2,$$

where the probability is over $m \xleftarrow{\$} \mathcal{M}$ and $r \xleftarrow{\$} \mathcal{R}$.

Remark 5 (An Easy Lower Bound). One can show that for any encoding function f mapping into the hypercube $[\mathsf{w}]^v$, we have $\mu_{\ell^2}(f) \geq 1/\mathsf{w}^v$, and that such a collision metric can also be achieved by a uniform encoding. The general lower bound can be seen using Jensen's inequality for the convex function $x \mapsto x^2$:

$$\begin{aligned} 1 = 1^2 &= \left(\sum_{x \in [\mathsf{w}]^v} \Pr_{m,r}\left[f(m,r) = x\right] \right)^2 \\ &= \mathsf{w}^{2v} \left(\frac{\sum_{x \in [\mathsf{w}]^v} \Pr_{m,r}\left[f(m,r) = x\right]}{w^v} \right)^2 \\ &\leq \mathsf{w}^{2v} \frac{\sum_{x \in [\mathsf{w}]^v} \Pr_{m,r}\left[f(m,r) = x\right]^2}{w^v} = \mathsf{w}^v \mu_{\ell^2}(f). \end{aligned}$$

In the following, we show that a high collision metric can be used to lower the security level of target collision resistance. Via Remark 4, this lowers the security level for Construction 1.

Lemma 1 (High Collision Metric Implies Insecurity)**.** *Let* w *and* v *be integers and consider the hypercube* $[\mathsf{w}]^v$*. Let* $f\colon \mathcal{M} \times \mathcal{R} \to [\mathsf{w}]^v$ *be an encoding function. If* f *is* ϵ*-secure with respect to target collision resistance, then we have*

$$\epsilon(1) \geq \mu_{\ell^2}(f) - 1/|\mathcal{M}|.$$

Proof. Consider the following adversary $\mathcal{A}$ in the target collision resistance game for encoding function f, as defined in Definition 7: first, $\mathcal{A}$ samples $m \xleftarrow{\$} \mathcal{M}$ and outputs it to the game. The game then samples $r \xleftarrow{\$} \mathcal{R}$ and gives it to $\mathcal{A}$. Now, $\mathcal{A}$ samples $(m^*, r^*) \xleftarrow{\$} \mathcal{M} \times \mathcal{R}$ and outputs it. We calculate the success probability of this adversary $\mathcal{A}$, where all probabilities are taken over $(m,r) \xleftarrow{\$} \mathcal{M} \times \mathcal{R}$ and $(m^*, r^*) \xleftarrow{\$} \mathcal{M} \times \mathcal{R}$:

$$\begin{aligned} \Pr\left[f(m,r) = f(m^*,r^*) \wedge m^* \neq m\right] &\geq \Pr\left[f(m,r) = f(m^*,r^*)\right] - \Pr\left[m^* = m\right] \\ &= \sum_{x \in [\mathsf{w}]^v} \Pr\left[f(m,r) = x \wedge f(m^*,r^*) = x\right] - 1/|\mathcal{M}| \\ &= \sum_{x \in [\mathsf{w}]^v} \Pr\left[f(m,r) = x\right] \cdot \Pr\left[f(m^*,r^*) = x\right] - 1/|\mathcal{M}| \\ &= \mu_{\ell^2}(f) - 1/|\mathcal{M}|. \end{aligned}$$

Clearly, our adversary has small constant running time. □

While Lemma 1 already shows that a low collision metric is necessary to get a high security level for Construction 1, it only considers a very specific attacker. One may be curious how the success probability of this adversary changes when we let it sample more candidates for (m^*, r^*). As we show in the next lemma, the success probability of such adversaries is still captured reasonably well by the collision metric.

Lemma 2 (High Collision Metric Implies Insecurity – More Generally)**.** *Let* w *and* v *be integers and consider the hypercube* $[\mathsf{w}]^v$*. Let* $f\colon \mathcal{M} \times \mathcal{R} \to [\mathsf{w}]^v$ *be an encoding function. If* f *is* ϵ*-secure with respect to target collision resistance, then we have*

$$\epsilon(t) \geq t \cdot \mu_{\ell^2}(f) - \sum_{x \in [\mathsf{w}]^v} \frac{t(t-1)}{2} p_x^3 - t/|\mathcal{M}| \quad \textit{for every } t \in \mathbb{N},$$

where $p_x := \Pr_{m,r}\left[f(m,r) = x\right]$ *for any* $x \in [\mathsf{w}]^v$.

Proof. Fix any $t \in \mathbb{N}$. We generalize the adversary $\mathcal{A}$ from the proof of Lemma 1 to an adversary $\mathcal{A}_t$ as follows: first, $\mathcal{A}_t$ samples $m \xleftarrow{\$} \mathcal{M}$ and outputs it to the game. The game then samples $r \xleftarrow{\$} \mathcal{R}$ and gives it to $\mathcal{A}_t$. Now, $\mathcal{A}_t$ samples t independent pairs $(m_i^*, r_i^*) \xleftarrow{\$} \mathcal{M} \times \mathcal{R}$ for $i \in [t]$. If there is an $i_0 \in [t]$ with $m_{i_0}^* \neq m$ and $f(m,r) = f(m_{i_0}^*, r_{i_0}^*)$, then $\mathcal{A}_t$ outputs $(m_{i_0}^*, r_{i_0}^*)$. Otherwise, it outputs $\perp$. Clearly, $\mathcal{A}_t$ runs in time t and if it outputs a pair, it breaks target collision resistance. What is left is to calculate the

probability that it finds such an i_0. We take the probabilities over $(m, r) \xleftarrow{\$} \mathcal{M} \times \mathcal{R}$ and $(m_i^*, r_i^*) \xleftarrow{\$} \mathcal{M} \times \mathcal{R}$ for $i \in [t]$. First, we can write the success probability as

$$\begin{aligned}\Pr\left[\exists i \in [t] \colon f(m,r) = f(m_i^*, r_i^*) \wedge m_i^* \neq m\right] &\geq \Pr\left[\exists i \in [t] \colon f(m,r) = f(m_i^*, r_i^*)\right] - \Pr\left[\exists i \in [t] \colon m_i^* = m\right] \\ &\geq \Pr\left[\exists i \in [t] \colon f(m,r) = f(m_i^*, r_i^*)\right] - t/|\mathcal{M}|.\end{aligned}$$

We focus on the first term now. Then, the first term becomes

$$\begin{aligned}\Pr\left[\exists i \in [t] \colon f(m,r) = f(m_i^*, r_i^*)\right] &= \sum_{x \in [\mathsf{w}]^v} p_x \cdot \Pr\left[\exists i \in [t] \colon f(m_i^*, r_i^*) = x\right] \\ &= \sum_{x \in [\mathsf{w}]^v} p_x \cdot (1 - \Pr\left[\forall i \in [t] \colon f(m_i^*, r_i^*) \neq x\right]) \\ &= \sum_{x \in [\mathsf{w}]^v} p_x \cdot (1 - (1-p_x)^t).\end{aligned}$$

We continue by using the Taylor expansion for $f(x) = (1-x)^\alpha$ with Lagrange remainder term[5]:

$$\begin{aligned}&\sum_{x \in [\mathsf{w}]^v} p_x \cdot (1 - (1-p_x)^t) \\ = &\sum_{x \in [\mathsf{w}]^v} p_x \cdot \left(1 - \left(1 - tp_x + \frac{t(t-1)}{2}p_x^2 - \frac{t(t-1)(t-2)}{6} \underbrace{\xi^{t-3}}_{\text{some value} > 0} p_x^4\right)\right) \\ = &\sum_{x \in [\mathsf{w}]^v} \left(tp_x^2 - \frac{t(t-1)}{2}p_x^3 + \frac{t(t-1)(t-2)}{6}\xi^{t-3}p_x^4\right) \\ \geq &\sum_{x \in [\mathsf{w}]^v} \left(tp_x^2 - \frac{t(t-1)}{2}p_x^3\right) \geq t \cdot \mu_{\ell^2}(f) - \sum_{x \in [\mathsf{w}]^v} \frac{t(t-1)}{2}p_x^3.\end{aligned}$$

□

3 Lower Bounds for the Cost of Encodings

In this section we derive a family of lower bounds for the cost of an encoding function f (Definition 5) whose collision metric $\mu_{\ell^2}(f)$ is at most μ.

3.1 Lower Bound Statement

The bounds can be computed for an arbitrary increasing[6] sequence $\{C_d\}_d$ of layer cost values. Precisely, this means we generalize our cost function from Definition 9 by assuming

$$\mathsf{cost}(f, m, r) = \widetilde{C_v} + C_{\mathrm{layer}(f(m,r))}.$$

The specific cost function from Definition 9 is obtained by setting $C_d = d$ for every d. Each lower bound is given for a particular hypercube $[\mathsf{w}]^v$ in the form $\mathsf{cost}(f) \geq \widetilde{C_v} + C_{\ell^2}(\mu, v, \mathsf{w})$. It should be interpreted as follows: any one-time chain-based signature scheme (Construction 1) with signatures of size v hash values and key generation time at most t_{max} oracle calls has verification cost at least $\widetilde{C_v} + \min_{\mathsf{w} \leq t_{max}/v+1} C_{\ell^2}(\mu, v, \mathsf{w})$. In our concrete evaluations of C_{ℓ^2} (Section 5) for fixed μ and v we have observed that it quickly approaches some lower bound as w grows so that bigger values of w bring no advantage. Note that in the following, the assumption $\mathsf{w}^{-v} \leq \mu_{\ell^2}(f)$ is without loss of generality, see Remark 5.

[5] Essentially it is for $0 < x < 1$: $(1-x)^\alpha = 1 - \alpha x + \frac{\alpha(\alpha-1)}{2}x^2 - \frac{\alpha(\alpha-1)(\alpha-2)}{6}x^3\xi^{t-3}$ for some $\xi \in [x; 1]$.

[6] If there is a non-decreasing sequence (with possible repetitions) then one should group layers with the same cost in order to use the theorem.

Theorem 1. *Let $f\colon \mathcal{M} \times \mathcal{R} \to [\mathsf{w}]^v$ be an encoding with collision metric $\mathsf{w}^{-v} \leq \mu_{\ell^2}(f) \leq \mu$, and let $\{C_d\}_{0 \leq d \leq v(\mathsf{w}-1)}$ be an increasing sequence of non-negative values. Then, for the encoding cost $\mathsf{cost}(f)$, defined as*

$$\mathsf{cost}(f) := \widetilde{C_v} + \mathop{\mathbb{E}}_{m,r}\left[C_{\mathrm{layer}(f(m,r))}\right]$$

the following lower bound holds:

- *If $\mu \geq 1/\ell_0$, then $\mathsf{cost}(f) \geq \widetilde{C_v} + C_0$.*
- *If $\mu = \mathsf{w}^{-v}$, then $\mathsf{cost}(f) = \widetilde{C_v} + \mathsf{w}^{-v} \cdot \sum_{0 \leq d \leq v(\mathsf{w}-1)} \ell_d C_d$.*
- *If $\mathsf{w}^{-v} < \mu < 1/\ell_0$, then*

$$\mathsf{cost}(f) \geq \widetilde{C_v} + C_{\ell^2}(\mu, v, \mathsf{w}) := \widetilde{C_v} + \min_{d_0} \sum_{0 \leq d \leq v(\mathsf{w}-1)} \ell_d \widehat{\mu}_d^{(d_0)} C_d,$$

where the minimum is taken over all d_0 for which $A \geq 0$ where

$$A := \frac{\left(\sum_{0 \leq d \leq d_0} \ell_d C_d\right)^2}{\ell_{[0:d_0]}^2} - \frac{\mu\left(\sum_{0 \leq d \leq d_0} \ell_d C_d\right)^2 - \sum_{0 \leq d \leq d_0} \ell_d C_d^2}{\mu \ell_{[0:d_0]}^2 - \ell_{[0:d_0]}}.$$

The values $\widehat{\mu}_d^{(d_0)}$ are determined from d_0 and A as

$$\widehat{\mu}_d^{(d_0)} = \max\left(\frac{\lambda_2^{(d_0)} - C_d}{2\lambda_1^{(d_0)}}, 0\right)$$

with

$$\lambda_1^{(d_0)} = \frac{\ell_{[0:d_0]}\sqrt{A}}{2},\ \lambda_2^{(d_0)} = \frac{\sum_{d \leq d_0} \ell_d C_d}{\ell_{[0:d_0]}} + \sqrt{A}.$$

Before we go into the proof of Theorem 1, we explain how it applies to chain-based signatures. Specifically, consider SIG in Construction 1, implemented on a hypercube $[\mathsf{w}]^v$ with an encoding function $f\colon \mathcal{M} \times \mathcal{R} \to [\mathsf{w}]^v$. Assume that SIG is ϵ-secure (Definition 8). Then, by Remark 4, we know that f is ϵ'-secure with respect to target collision resistance (Definition 7) for $\epsilon' \leq \epsilon$. So, using Lemma 1, we get

$$\mu_{\ell^2}(f) - 1/|\mathcal{M}| \leq \epsilon'(1) \leq \epsilon(1).$$

Now, fixing any μ, there are two cases: either, $\mu_{\ell^2}(f) > \mu$. Then, we get $\epsilon(1) > \mu - 1/|\mathcal{M}|$. Otherwise, our lower bound from Theorem 1 applies. We get the following corollary.

Corollary 1 (Informal)**.** *Consider* SIG *in Construction 1, implemented on a hypercube $[\mathsf{w}]^v$ with an encoding function $f\colon \mathcal{M} \times \mathcal{R} \to [\mathsf{w}]^v$. Fix any μ. Then, we are in one of two cases:*

1. *Either, $\mu_{\ell^2}(f) > \mu$. Then, if* SIG *is ϵ-secure, we have $\epsilon(1) > \mu - 1/|\mathcal{M}|$.*
2. *Or, $\mu_{\ell^2}(f) \leq \mu$. Then, our lower bound from Theorem 1 yields a lower bound on the verification cost of* SIG*.*

3.2 Lower Bound Proof

We now turn to the proof of Theorem 1. First, we give an outline, before going into detail.

Proof Outline

Our proof consists of four steps, summarized as follows:

> **Step 1.** First, we reduce the lower bound problem to an optimization problem for non-negative functions defined on the hypercube, which we call *density functions*.
>
> **Step 2.** We show that the optimal density functions are constant on each hypercube layer.
>
> **Step 3.** We show that the optimal density functions are non-zero on the hypercube layers from 0 to some d_0, and zero on all others.
>
> **Step 4.** We show that the optimal density function on its non-zero part is an affine function of the layer cost C_d, with coefficients being determined from a quadratic equation, and which is solvable only for certain d_0. When it is solvable, its two solutions yield a minimum and a maximum density functions, of which the former is taken. The final lower bound is the minimum over all solutions for different d_0.

Step 1: Reduction to Density Functions

We first note that the bound in the case $\mu \geq 1/\ell_0$ is trivial as C_0 is the smallest layer cost, which means that $\mathsf{cost}(f) = \widetilde{C_v} + \mathbb{E}_{m,r}\left[C_{\mathrm{layer}(f(m,r))}\right] \geq \widetilde{C_v} + \mathbb{E}_{m,r}\left[C_0\right] = \mathsf{cost}(f) \geq \widetilde{C_v} + C_0$. It is also trivial to verify that this bound is actually achieved by the encoding that maps uniformly into layer 0 as its collision metric is equal to $1/\ell_0$.

The rest of the proof is devoted to the case $1/\mathsf{w}^v \leq \mu < 1/\ell_0$. Let us associate with each encoding f the function $\widehat{\mu}_f$, which we call *density function*:

$$\widehat{\mu}_f \colon [\mathsf{w}]^v \to \mathbb{R}, \quad \widehat{\mu}_f(x) = \frac{\left|\{(m,r) \in (\mathcal{M},\mathcal{R}) \mid f(m,r) = x\}\right|}{|\mathcal{M}| \cdot |\mathcal{R}|}.$$

That is, $\widehat{\mu}_f(x)$ tells the fraction[7] of the domain of f that maps into x. Then, the cost $\mathsf{cost}(f)$ can be expressed as

$$\mathsf{cost}(f) = \widetilde{C_v} + \sum_{x \in [\mathsf{w}]^v} \widehat{\mu}_f(x) C_{\mathrm{layer}(x)}.$$

Consequently, the cost function $\mathsf{cost}(f)$ can be computed from the density function alone. Thus, the lower bound on cost is yielded by a solution to the following *optimization problem* for density functions:

$$\sum_{x \in [\mathsf{w}]^v} \widehat{\mu}(x) C_{\mathrm{layer}(x)} \xrightarrow{\widehat{\mu}} \min, \tag{1}$$

where the minimum is taken over all *non-negative* functions $\widehat{\mu}$ satisfying *boundary conditions*:

$$\text{Collision metric:} \quad \sum_{x \in [\mathsf{w}]^v} \widehat{\mu}(x)^2 \leq \mu;$$

$$\text{Domain coverage:} \quad \sum_{x \in [\mathsf{w}]^v} \widehat{\mu}(x) = 1.$$

For any such non-negative function, let us denote

$$\mathsf{cost}(\widehat{\mu}) := \sum_{x \in [\mathsf{w}]^v} \widehat{\mu}(x) C_{\mathrm{layer}(x)}.$$

Note that the function cost maps $\mathbb{R}^{\mathsf{w}^v}$ to $\mathbb{R}$. It is continuous and the boundary conditions form a *compact* set. Therefore, by the Weierstrass theorem an optimal solution to the problem (1) *exists*.

[7] Whereas practically the density function is rational-valued, for the purpose of analysis and without loss of generality we allow it to take *real* non-negative values.

Now we can finish the case $\mu = 1/\mathsf{w}^v$. Expanding the domain coverage equation we have

$$1^2 = \left(\sum_{x\in[\mathsf{w}]^v} \widehat{\mu}(x)\right)^2 = \mathsf{w}^{2v}\left(\frac{\sum_{x\in[\mathsf{w}]^v}\widehat{\mu}(x)}{\mathsf{w}^v}\right)^2 \le \mathsf{w}^{2v}\frac{\sum_{x\in[\mathsf{w}]^v}\widehat{\mu}(x)^2}{\mathsf{w}^v} \le \mu\mathsf{w}^v = 1,$$

where we have used Jensen's inequality[8]. Thus the inequality is an equation, which is possible only if all elements of the sum are identical, i.e., $\widehat{\mu}(x) = 1/\mathsf{w}^v$ for every x. From here we get that

$$\mathsf{cost}(f) = \widetilde{C_v} + \frac{\sum_{0\le d\le v(\mathsf{w}-1)}\ell_d C_d}{\mathsf{w}^v}.$$

The rest of the proof deals with the case $1/\mathsf{w}^v < \mu < 1/\ell_0$.

Step 2: Optimal Density Is Layer-Constant

Let us show that for any function $\widehat{\mu}'$, which satisfies boundary conditions, there exists a function $\widehat{\mu}$ with $\mathsf{cost}(\widehat{\mu}) = \mathsf{cost}(\widehat{\mu}')$, which is constant on each layer. Indeed, suppose there is a layer $\mathcal{L}_d$ on which $\widehat{\mu}'$ is not constant. Let $\widehat{\mu}$ take exactly the same values as $\widehat{\mu}'$ except that it takes its average values on $\mathcal{L}_d$:

$$\forall x \in \mathcal{L}_d\colon\ \ \widehat{\mu}(x) = \sum_{y\in\mathcal{L}_d}\frac{\widehat{\mu}'(y)}{\ell_d}.$$

It is easy to see that $\sum_{x\in[\mathsf{w}]^v}\widehat{\mu}(x) = \sum_{x\in[\mathsf{w}]^v}\widehat{\mu}'(x) = 1$ and $\mathsf{cost}(\widehat{\mu}) = \mathsf{cost}(\widehat{\mu}')$, whereas for the collision metric we obtain

$$\begin{aligned}\sum_{x\in[\mathsf{w}]^v}\widehat{\mu}(x)^2 &= \sum_{x\in[\mathsf{w}]^v}\widehat{\mu}'(x)^2 + \left(\sum_{x\in\mathcal{L}_d}\widehat{\mu}(x)^2 - \sum_{x\in\mathcal{L}_d}\widehat{\mu}'(x)^2\right)\\ &\le \mu + \left(\ell_d\left(\sum_{x\in\mathcal{L}_d}\frac{\widehat{\mu}'(x)}{\ell_d}\right)^2 - \sum_{x\in\mathcal{L}_d}\widehat{\mu}'(x)^2\right) \le \mu,\end{aligned}$$

where in the last step we have used Jensen's inequality for the convex function $x \mapsto x^2$ again. Therefore, the boundary conditions remain satisfied. We can repeat this process for every layer. Eventually, we reduce the minimization problem to the search among functions that take a single non-negative value on each layer $\mathcal{L}_d$, which we denote by $\widehat{\mu}_d$ for $0 \le d \le v(\mathsf{w}-1)$:

$$\sum_{0\le d\le v(\mathsf{w}-1)}\widehat{\mu}_d\ell_d C_d \xrightarrow{\{\widehat{\mu}_d\}} \min; \tag{2}$$

$$\text{Collision metric:}\qquad \sum_{0\le d\le v(\mathsf{w}-1)}\widehat{\mu}_d^2\ell_d \le \mu; \tag{3}$$

$$\text{Domain coverage:}\qquad \sum_{0\le d\le v(\mathsf{w}-1)}\widehat{\mu}_d\ell_d = 1 \tag{4}$$

$$\text{Non-negativity:}\qquad \widehat{\mu}_d \ge 0,\quad 0 \le d \le v(\mathsf{w}-1). \tag{5}$$

Step 3: Optimal Density is Cut Off

We now show that for the optimal solution all positive values $\widehat{\mu}_d$ must be in the first layers, or, in other words, the following implication must hold for every $d'' > d'$:

$$\widehat{\mu}_{d'} = 0 \Longrightarrow \widehat{\mu}_{d''} = 0. \tag{6}$$

Suppose otherwise for a function $\widehat{\mu}'$, which satisfies boundary conditions, there are some $d' < d''$ such that

$$\widehat{\mu}'_{d'} = 0,\quad \widehat{\mu}'_{d''} = a > 0.$$

[8]We use it for the convex function $x \mapsto x^2$. Then, Jensen's inequality states $(\sum_i a_i/n)^2 \le (\sum_i a_i^2)/n$.

Note that $a < 1$ as otherwise the collision metric would be at least 1 – contradictory to our restriction on μ. Then let us consider a new function $\widehat{\mu}$ that is equal to $\widehat{\mu}'$ on all layers except that

$$\widehat{\mu}_{d'} = \frac{a}{2\ell_{d'}}, \quad \widehat{\mu}_{d''} = a - \frac{a}{2\ell_{d''}}.$$

We then obtain that the cost function *decreases* (2)

$$\mathsf{cost}(\widehat{\mu}) = \mathsf{cost}(\widehat{\mu}') + \underbrace{aC_{d'}/2 - aC_{d''}/2}_{<0 \text{ as } C_d \text{ increases with } d} < \mathsf{cost}(\widehat{\mu}').$$

For the domain coverage (4):

$$\sum_{0 \le d \le v(\mathsf{w}-1)} \widehat{\mu}_d \ell_d = \sum_{0 \le d \le v(\mathsf{w}-1)} \widehat{\mu}'_d \ell_d = 1.$$

And for the collision metric (3):

$$\sum_{0 \le d \le v(\mathsf{w}-1)} \widehat{\mu}_d^2 \ell_d = \sum_{0 \le d \le v(\mathsf{w}-1)} \widehat{\mu}_d^{'2} \ell_d + (\widehat{\mu}_{d'}^2 - \widehat{\mu}_{d'}^{'2})\ell_{d'} + (\widehat{\mu}_{d''}^2 - \widehat{\mu}_{d''}^{'2})\ell_{d''} \le$$

$$\le \mu + \ell_{d'}\left(\frac{a^2}{4\ell_{d'}^2} - 0\right) + \ell_{d''}\left(\left(a - \frac{a}{2\ell_{d''}}\right)^2 - a^2\right) = \mu + \frac{a^2}{4\ell_{d'}} - a^2 + \frac{a^2}{4\ell_{d''}} \le$$

$$\le \mu + a^2 \underbrace{\left(\frac{1}{4\ell_{d'}} - 1 + \frac{1}{4\ell_{d''}}\right)}_{<0 \text{ as } \ell_{d'}, \ell_{d''} \ge 1 > a} < \mu$$

Therefore, the boundary conditions remain satisfied and thus $\widehat{\mu}'$ is not optimal. Consequently, we can reduce the minimization problem to the search among functions that are non-zero on layers from 0 to some d_0.

Step 4: Analytical Solution

In order to solve the optimization problem (2-5) analytically, we get rid of inequalities (5) by replacing variables

$$\widehat{\mu}_d \to \nu_d^2$$

and turn inequality (3) into an equation by introducing a new variable ν_{-1}:

$$\sum_{0 \le d \le v(\mathsf{w}-1)} \nu_d^2 \ell_d C_d \xrightarrow{\{\nu_d\}} \min; \tag{7}$$

$$\nu_{-1}^2 + \sum_{0 \le d \le v(\mathsf{w}-1)} \nu_d^4 \ell_d = \mu; \tag{8}$$

$$\sum_{0 \le d \le v(\mathsf{w}-1)} \nu_d^2 \ell_d = 1. \tag{9}$$

Candidate solutions to the system (7)-(9) are found by composing Lagrangian $\mathcal{L}(\nu_{-1}, \nu_0, \ldots, \lambda_1, \lambda_2)$, which is given as

$$\sum_{0 \le d \le v(\mathsf{w}-1)} \ell_d \nu_d^2 C_d + \lambda_1 \left(\nu_{-1}^2 + \sum_{0 \le d \le v(\mathsf{w}-1)} \ell_d \nu_d^4 - \mu\right) + \lambda_2 \left(1 - \sum_{0 \le d \le v(\mathsf{w}-1)} \ell_d \nu_d^2\right)$$

and solving the system of equations

$$\frac{d\mathcal{L}}{d\nu_d} = 0, \quad -1 \le d \le v(\mathsf{w}-1)$$
$$\frac{d\mathcal{L}}{d\lambda_1} = 0;$$
$$\frac{d\mathcal{L}}{d\lambda_2} = 0.$$

This can be rewritten as

$$2\lambda_1\nu_{-1} = 0, \tag{10}$$

$$2\ell_d C_d\nu_d + 4\lambda_1\ell_d\nu_d^3 - 2\lambda_2\ell_d\nu_d = 0, \quad 0 \le d \le v(\mathsf{w}-1) \tag{11}$$

$$\nu_{-1}^2 + \sum_{0\le d\le v(\mathsf{w}-1)} \ell_d\nu_d^4 = \mu; \tag{12}$$

$$\sum_{0\le d\le v(\mathsf{w}-1)} \ell_d\nu_d^2 = 1 \tag{13}$$

The first equation yields two options: $\lambda_1 = 0$ and $\lambda_1 \neq 0$. In the first case Eq. (11) transforms into

$$\nu_d\ell_d(C_d - \lambda_2) = 0,$$

which holds for every $d \ge 0$. As $\nu_0 \neq 0$ from Step 3, we effectively have $\lambda_2 = C_0$ and $\nu_d = 0$ for all $d > 0$ since C_d is increasing. Therefore from (3) we have $\mu \ge 1/\ell_0$, which contradicts our restriction on μ.

Therefore, $\lambda_1 \neq 0$ and thus $\nu_{-1} = 0$ from (10). Then (11) transforms to

$$\nu_d^2 \in \left\{0, \frac{\lambda_2 - C_d}{2\lambda_1}\right\}, \quad \forall d \ge 0.$$

Due to (6), we effectively have for some d_0

$$\nu_d^2 = \begin{cases} \frac{\lambda_2 - C_d}{2\lambda_1}, & \text{for } 0 \le d \le d_0; \\ 0, & \text{for } d > d_0 \end{cases} \tag{14}$$

So from (13) we have

$$\sum_{0\le d\le d_0} \ell_d \frac{\lambda_2 - C_d}{2\lambda_1} = 1,$$

which turns into an equation for λ_1:

$$\lambda_1 = \frac{\sum_{0\le d\le d_0} \ell_d\,(\lambda_2 - C_d)}{2} = \lambda_2\ell_{[0:d_0]}/2 - \frac{\sum_{0\le d\le d_0} \ell_d C_d}{2}. \tag{15}$$

Now we rewrite (12) as

$$\sum_{0\le d\le d_0} \ell_d(\lambda_2 - C_d)^2 = 4\mu\lambda_1^2$$

and substitute λ_1 from (15):

$$\sum_{0\le d\le d_0} \ell_d(\lambda_2 - C_d)^2 = \mu\left(\lambda_2^2\ell_{[0:d_0]}^2 - 2\ell_{[0:d_0]}\lambda_2 \sum_{0\le d\le d_0} \ell_d C_d + \left(\sum_{0\le d\le d_0} \ell_d C_d\right)^2\right).$$

Grouping the coefficients we get

$$\lambda_2^2\left(\ell_{[0:d_0]} - \mu\ell_{[0:d_0]}^2\right) + \lambda_2\left(-2\sum_{0\le d\le d_0} \ell_d C_d + 2\mu\ell_{[0:d_0]} \sum_{0\le d\le d_0} \ell_d C_d\right)$$
$$+ \sum_{0\le d\le d_0} \ell_d C_d^2 - \mu\left(\sum_{0\le d\le d_0} \ell_d C_d\right)^2 = 0. \tag{16}$$

Let us show that if the equation holds then the coefficient at λ_2^2 must be non-zero. Suppose otherwise, so that $\mu = \ell_{[0:d_0]}$. Then both coefficients in Eq.(16) become 0 and we have for the free term

$$\sum_{0\le d\le d_0} \ell_d C_d^2 = \frac{1}{\ell_{[0:d_0]}}\left(\sum_{0\le d\le d_0} \ell_d C_d\right)^2$$

or equivalently

$$\left(\sum_{0\le d\le d_0}\ell_d\right)\left(\sum_{0\le d\le d_0}\ell_d C_d^2\right)=\left(\sum_{0\le d\le d_0}\ell_d C_d\right)^2. \tag{17}$$

Now, we recall the Hölder / Cauchy-Schwarz inequality for positive reals:

$$\sum_i x_i y_i \le \left(\sum_i x_i^2\right)^{1/2}\left(\sum_i y_i^2\right)^{1/2}.$$

This is an equality only if the sequence $\{x_i^2\}$ is proportional to $\{y_i^2\}$, which in our case ($x_i^2 := \ell_i,\ y_i^2 := \ell_i C_i^2$) implies that $\{\ell_d\}$ is proportional to $\{\ell_d C_d\}$.

This is a contradiction since C_d is increasing, so for an optimal function it should hold that $\mu \neq \ell_{[0:d_0]}$. Thus we can divide Eq. (16) by $\ell_{[0:d_0]}(1-\mu\ell_{[0:d_0]})$ and get

$$\lambda_2^2 - 2\lambda_2\frac{\sum_{0\le d\le d_0}\ell_d C_d}{\ell_{[0:d_0]}} + \frac{\mu\left(\sum_{0\le d\le d_0}\ell_d C_d\right)^2 - \sum_{0\le d\le d_0}\ell_d C_d^2}{\mu\ell_{[0:d_0]}^2 - \ell_{[0:d_0]}} = 0.$$

This is a quadratic equation on λ_2. Suppose it has solutions for some d_0. Then, the solutions yield two extremum solutions for the optimization problem: one that maximizes the cost function, and the one that minimizes it. The two solutions for λ_2 are

$$\lambda_2 = \frac{\sum_{d\le d_0}\ell_d C_d}{\ell_{[0:d_0]}} \pm \sqrt{A},$$

where

$$A = \frac{\left(\sum_{d\le d_0}\ell_d C_d\right)^2}{\ell_{[0:d_0]}^2} - \frac{\mu\left(\sum_{d\le d_0}\ell_d C_d\right)^2 - \sum_{d\le d_0}\ell_d C_d^2}{\mu\ell_{[0:d_0]}^2 - \ell_{[0:d_0]}}.$$

Each of them determines λ_1 via Eq.(15):

$$\lambda_1 = \pm\frac{\ell_{[0:d_0]}\sqrt{A}}{2}.$$

We are able to determine which solution corresponds to the minimum point of the cost function. Indeed, if λ_1 is positive then $\lambda_2 \ge C_d$ from (14), i.e. ν_d decreases with d. In the other case, if λ_1 is negative then $\lambda_2 \le C_d$ from (14), i.e. ν_d increases with d. Using the same argument as in Step 3, we obtain that the former case corresponds to the minumum case, and the latter case – to the maximum case. Therefore,

$$\lambda_1 = \frac{\ell_{[0:d_0]}\sqrt{A}}{2}, \qquad \lambda_2 = \frac{\sum_{d\le d_0}\ell_d C_d}{\ell_{[0:d_0]}} + \sqrt{A}.$$

Then we determine all ν_d (and thus $\widehat{\mu}_d$) via Eq.(14). This ends the proof.

4 Novel Encodings

In this section we present our constructions. Namely, we present explicit encodings that almost achieve the lower bounds derived in Section 3. Our first encoding (Construction 2) builds an encoding from the optimal density function described in Theorem 1, and then makes it incomparable by adding a checksum. However, the overhead incurred by the checksum chain verification is non-negligible, so we suggest two more constructions. The second encoding scheme (Construction 3) builds an encoding from the density function that is non-zero on only a few layers, thus enabling a 1-chain checksum. The third scheme (Construction 4) uses a mapping into just a single layer of the hypercube. In all schemes, we implement a non-uniform mapping into a hypercube by combining a random oracle H and a non-uniform function Ψ. To analyze security of these encodings, we first present a lemma (Lemma 3) that relates the target collision resistance of $\Psi \circ \mathsf{H}$ to the collision metric of Ψ. We then use this lemma when presenting our encodings based on such functions Ψ. In the final part of this section, we explain how to efficiently implement the functions Ψ for our encodings.

4.1 Random Oracles with Postprocessing

As already mentioned, our encodings will apply a function Ψ to the output of a random oracle H. In the following, we prove a generic lemma that relates the target collision resistance of this combined function to the collision metric of Ψ. We will use this lemma later to show security of our encodings.

Lemma 3. *Let $\widehat{\mu}\colon \mathbb{R} \to [0,1]$ be a function. Let $\mathsf{H}\colon \mathcal{M} \times \mathcal{R} \to \mathcal{X}$ be a random oracle, and let $\Psi\colon \mathcal{X} \to \mathcal{Y}$ be a function such that for every $y \in \mathcal{Y}$, we have*

$$\Pr_{x \xleftarrow{\$} \mathcal{X}}[\Psi(x) = y] = \widehat{\mu}(y).$$

For any algorithm $\mathcal{A}$, define $\epsilon_{\mathcal{A}}$ as that makes at most t queries to H, we have

$$\epsilon_{\mathcal{A}} := \Pr\left[\Psi(\mathsf{H}(m,r)) = \Psi(\mathsf{H}(m^*,r^*)) \wedge m \neq m^* \;\middle|\; \begin{array}{c}(m,St) \leftarrow \mathcal{A}^{\mathsf{H}},\ r \xleftarrow{\$} \mathcal{R},\\ (m^*,r^*) \leftarrow \mathcal{A}^{\mathsf{H}}(St,r)\end{array}\right].$$

Intuitively, $\epsilon_{\mathcal{A}}$ denotes the advantage of $\mathcal{A}$ against target collision resistance of $\Psi \circ \mathsf{H}$, in the random oracle model. Then, for any $\mathcal{A}$ that makes at most t queries to H, we have

$$\epsilon_{\mathcal{A}} \leq \frac{t}{|\mathcal{R}|} + t\left(\sum_{y \in \mathcal{Y}} \widehat{\mu}^2(y)\right).$$

Proof. Let $\mathcal{A}$ be an adversary as in the lemma. Assume that $\mathcal{A}$ makes (at most) t_0 queries to H before outputting m, and denote these queries by $((m_i^1, r_i^1))_i$. Then, $\mathcal{A}$ gets r and makes (at most) $t - t_0$ queries to H, denoted by $((m_j^2, r_j^2))_j$. Without loss of generality, we assume that all queries are fresh, and one of them is the final output (m^*, r^*). To bound $\epsilon_{\mathcal{A}}$, we consider three cases, and bound the probability of winning in each case separately:

Case 1. (m,r) was queried in Step 1 of the game.

Case 2. (m,r) was not queried in Step 1 of the game, but it forms a collision with some query made in Step 1: there exists $i \in [t_0]$ with $\Psi(\mathsf{H}(m,r)) = \Psi(\mathsf{H}(m_i^1, r_i^1))$ and $m \neq m_i^1$.

Case 3. (m,r) was not queried in Step 1 of the game, but it forms a collision with some query made in Step 2: there exists $j \in [t - t_0]$ with $\Psi(\mathsf{H}(m,r)) = \Psi(\mathsf{H}(m_j^2, r_j^2))$ and $m \neq m_j^2$.

Case 1. The probability of this case is clearly upper bounded by the probability that $r = r_i^1$ for some i, which is at most $t/|\mathcal{R}|$.

Case 2. Fix any $i \in [t_0]$. Since the queries $(m,r), (m_i^1, r_i^1)$ are distinct, the values $\mathsf{H}(m,r), \mathsf{H}(m_i^1, r_i^1)$ are uniformly and independently distributed. For every $y \in \mathcal{Y}$, the probability that both (m_i^1, r_i^1) and (m,r) queries result in y is exactly:

$$\Pr_{\mathsf{H}}[\Psi(\mathsf{H}(m,r)) = y] = \Pr_{\mathsf{H}}\left[\Psi(\mathsf{H}(m_i^1, r_i^1)) = y\right] = \widehat{\mu}(y).$$

Therefore, we can upper bound the probability in this case as

$$\begin{aligned}\Pr\left[\exists i \in [t_0]\colon \Psi(\mathsf{H}(m,r)) = \Psi(\mathsf{H}(m_i^1, r_i^1)),\ m \neq m_i^1\right] \leq \\ \leq \sum_{y \in \mathcal{Y}} \sum_{1 \leq i \leq t_0} \Pr_{\mathsf{H}}\left[\Psi(\mathsf{H}(m_i^1, r_i^1)) = y\right] \Pr_{\mathsf{H}}[\Psi(\mathsf{H}(m,r)) = y] = \sum_{y \in \mathcal{Y}} t_0 \widehat{\mu}^2(y).\end{aligned}$$

Case 3. This is exactly as in Case 2. Namely, for every $y \in \mathcal{Y}$, the probability that (m,r) and (m_j^2, r_j^2) are mapped to y is

$$\Pr_{\mathsf{H}}[\Psi(\mathsf{H}(m,r)) = y] = \Pr_{\mathsf{H}}\left[\Psi(\mathsf{H}(m_j^2, r_j^2)) = y\right] = \widehat{\mu}(y).$$

Therefore, we can upper bound the probability in this case as

$$\begin{aligned}\Pr\left[\exists j \in [t - t_0]\colon \mathsf{H}(m,r) = \mathsf{H}(m_j^2, r_j^2),\ m \neq m_j^2\right] \leq \\ \leq \sum_{y \in \mathcal{Y}} \sum_{1 \leq j \leq t - t_0} \Pr_{\mathsf{H}}\left[\Psi(\mathsf{H}(m_j^2, r_j^2)) = y\right] \Pr_{\mathsf{H}}[\Psi(\mathsf{H}(m,r)) = y] = \sum_{y \in \mathcal{Y}} (t - t_0) \widehat{\mu}^2(y).\end{aligned}$$

Adding the three cases together, we obtain the lemma statement. □

4.2 Our Encodings

Equipped with Lemma 3, we can present our encodings. Each of them will make use of a function Ψ, which we specify abstractly by specifying the density function $\widehat{\mu}$. We then show incomparability (Definition 6) and make use of Lemma 3 to show target collision resistance. Via Remark 4, we then get that the resulting one-time signature scheme is secure. Note that at this point, it is not clear that such functions Ψ for the given $\widehat{\mu}$ even exist. Looking ahead, in Section 4.3, we will show that these functions exist by efficiently implementing them.

As indicated by Theorem 1, the lower bound is yielded by a density function $\widehat{\mu}$ that is non-zero on layers from 0 to some d_0, taking value $\widehat{\mu}_d$ on layer d. Our first encoding uses this density function to map to those layers non-uniformly following the density function. We then attach to each layer a checksum to ensure incomparability (Definition 6). The hypercube size is taken to contain at least 2^λ vertices, where λ is a security parameter.

Construction 2 (TLFC: Top Layers with a Full Checksum)**.** *Let $\mathcal{M}$ and $\mathcal{R}$ be message and randomness spaces, respectively. Let λ be a parameter. Before we present the encoding, fix the following parameters:*

- *Select w, v such that $\mathsf{w}^v > 2^\lambda$ and set $\mu := 2^{-\lambda}$.*
- *Apply Theorem 1 to the hypercube $[\mathsf{w}]^v$ and μ with $C_d := d$. As a result, find values $d_0, \{\widehat{\mu}_d\}$ according to Theorem 1.*
- *Consider a random oracle $\mathsf{H}\colon \mathcal{M} \times \mathcal{R} \to \{0, 1, \ldots, \mathsf{w}^v - 1\}$*
- *Let $\Psi\colon \{0, 1, \ldots, \mathsf{w}^v - 1\} \to [\mathsf{w}^v]$ be a function such that for $z \xleftarrow{\$} \{0, 1, \ldots, \mathsf{w}^v - 1\}$, we have*

$$\forall\, x \in [\mathsf{w}]^v\colon \ \Pr_z[\Psi(z) = x] = \widehat{\mu}_{\mathrm{layer}(x)}.$$

- *Set $v' := \lceil \log_{\mathsf{w}}(d_0 + 1) \rceil$.*

With this, we now present our encoding $f\colon \mathcal{M} \times \mathcal{R} \to [\mathsf{w}]^{v+v'}$:

1. *Take as input $(m, r) \in \mathcal{M} \times \mathcal{R}$.*
2. *Set $x := \Psi(\mathsf{H}(m, r))$. Note that $x \in [\mathsf{w}]^v$. Set $d := \mathrm{layer}(x)$.*
3. *Interpret d as a base-w integer $(b_1, b_2, \ldots, b_{v'}) \in [\mathsf{w}]^{v'}$.*
4. *Output $(x_1, x_2, \ldots, x_v, b_1, b_2, \ldots, b_{v'})$ as the encoding.*

In the following, we show incomparability and target collision resistance of the encoding f specified in Construction 2.

Lemma 4. *Consider the encoding $f\colon \mathcal{M} \times \mathcal{R} \to [\mathsf{w}]^{v+v'}$ specified in Construction 2. Then, f is incomparable. Further, f is ϵ-secure against target collision resistance, where for any t, we have*

$$\epsilon(t) \leq t \cdot \left(\frac{1}{|\mathcal{R}|} + 2^{-\lambda} \right).$$

Proof. We start with incomparability. Consider two distinct outputs (x, b) and (x', b') with $x, x' \in [\mathsf{w}]^v$ and $b, b' \in [\mathsf{w}]^{v'}$ of the encoding f. If $x = x'$, then they are in the same layer and so we have $b = b'$, which contradicts $(x, b) \neq (x', b')$. So, we can assume that $x \neq x'$. Now, assume towards contradiction that $(x, b) \leq (x', b')$ (see Definition 2). Then, in particular, we have $x \leq x'$. As x and x' are distinct, this means that they are in different layers d and d' respectively, with $d > d'$. This implies that $b \not\leq b'$, which contradicts $(x, b) \leq (x', b')$.

It remains to show target collision resistance. Note that this follows directly from Lemma 3. In particular, a collision on the encodings implies a collision on the truncated encodings (without the b part). From the proof of Theorem 1, one can see that the collision metric induced by the $d_0, \{\widehat{\mu}_d\}$ is at most $\mu = 2^{-\lambda}$. □

Our first construction is arguably non-optimal: we use $v' = \lceil \log_{\mathsf{w}}(d_0+1) \rceil$ positions for the checksum, whereas certain values occur far less often than other ones. To address this, we suggest two other constructions, which suffer from a smaller overhead due to the checksum. Our next construction maps $\mathcal{M} \times \mathcal{R}$ again into some d_0+1 consecutive layers of the hypercube, but uses a single chain for the checksum. Strictly speaking, the construction does not fit into the framework of mapping into a hypercube (as it maps into $[\mathsf{w}]^v \times [d_0+1]$). However, it is trivial to generalize all definitions, constructions, and proofs from Section 2 to this setting.

Construction 3 (TL1C: Top Layers with a 1-Chain Checksum)**.** *Let $\mathcal{M}$ and $\mathcal{R}$ be message and randomness spaces, respectively. Let λ be a parameter. Before we present the encoding, fix the following parameters:*

- *Select w, v such that $\mathsf{w}^v > 2^\lambda$.*
- *Find the minimum d_0 such that $\ell_{[0:d_0]} \geq 2^\lambda$.*
- *Consider a random oracle $\mathsf{H}\colon \mathcal{M} \times \mathcal{R} \to \{0,1,\dots,\mathsf{w}^v-1\}$*
- *Let $\Psi\colon \{0,1,\dots,\mathsf{w}^v-1\} \to [\mathsf{w}^v]$ be a function such that for $z \xleftarrow{\$} \{0,1,\dots,\mathsf{w}^v-1\}$, we have*

$$\forall\, x \in [\mathsf{w}]^v\colon \ \Pr_z[\Psi(z)=x] = \widehat{\mu}(x),$$

where $\widehat{\mu}(x) = \begin{cases} 1/\ell_{[0:d_0]}, & \text{if } \mathrm{layer}(x) \leq d_0, \\ 0, & \text{if } \mathrm{layer}(x) > d_0. \end{cases}$

With this, we now present our encoding $f\colon \mathcal{M} \times \mathcal{R} \to [\mathsf{w}]^v \times [d_0+1]$:

1. *Take as input $(m,r) \in \mathcal{M} \times \mathcal{R}$.*
2. *Set $x := \Psi(\mathsf{H}(m,r))$. Note that $x \in [\mathsf{w}]^v$ and $\mathrm{layer}(x) \leq d_0$.*
3. *Set $b := \mathrm{layer}(x)+1$ as a checksum.*
4. *Output $(x_1,\dots,x_v,b)$ as the encoding.*

Lemma 5. *Consider the encoding $f\colon \mathcal{M} \times \mathcal{R} \to [\mathsf{w}]^{v+v'}$ specified in Construction 3. Then, f is incomparable. Further, f is ϵ-secure against target collision resistance, where for any t, we have*

$$\epsilon(t) \leq t \cdot \left(\frac{1}{|\mathcal{R}|} + 2^{-\lambda} \right).$$

Proof. Incomparability follows with the same arguments as for Construction 2. For target collision resistance, we can again apply Lemma 3. In order to do so, we compute

$$\sum_{x \in [\mathsf{w}]^v} \widehat{\mu}^2(x) = \sum_{0 \leq d \leq d_0} \ell_d \frac{1}{\ell^2_{[0:d_0]}} = \frac{\ell_{[0:d_0]}}{\ell^2_{[0:d_0]}} = \frac{1}{\ell_{[0:d_0]}} \leq 2^{-\lambda}.$$

□

Our third construction maps to a single layer of the hypercube so that no checksum is needed. Conceptually, this is similar to the constant sum encoding that has been studied before. However, instead of mapping into the largest layer of a small hypercube, we can map into a top layer of a larger hypercube.

Construction 4 (TSL: Top Single Layer)**.** *Let $\mathcal{M}$ and $\mathcal{R}$ be message and randomness spaces, respectively. Let λ be a parameter. Before we present the encoding, fix the following parameters:*

1. *Select w, v such that*[9] *$\mathsf{w}^v > 2^{\lambda + \log_4 \lambda}$.*
2. *Find the minimum d_0 such that $\ell_{d_0} \geq 2^\lambda$.*
3. *Consider a random oracle $\mathsf{H}\colon \mathcal{M} \times \mathcal{R} \to \{0,1,\dots,\mathsf{w}^v-1\}$*

[9] The extra term $\log_4 \lambda$ will ensure that a suitable d_0 exists.

4. *Let* $\Psi\colon \{0,1,\ldots,\mathsf{w}^v-1\} \to [\mathsf{w}^v]$ *be a function such that for* $z \xleftarrow{\$} \{0,1,\ldots,\mathsf{w}^v-1\}$*, we have*

$$\forall\, x \in [\mathsf{w}]^v\colon \quad \Pr_z[\Psi(z)=x] = \widehat{\mu}(x),$$

where $\widehat{\mu}(x) = \begin{cases} 1/\ell_{d_0}, & \text{if } \mathrm{layer}(x) = d_0, \\ 0, & \text{if } \mathrm{layer}(x) \neq d_0. \end{cases}$

With this, we now present our encoding $f\colon \mathcal{M}\times\mathcal{R} \to [\mathsf{w}]^v$

1. *Take as input* $(m,r) \in \mathcal{M}\times\mathcal{R}$*.*
2. *Set* $x := \Psi(\mathsf{H}(m,r))$*. Note that* $x \in [\mathsf{w}]^v$ *and* $\mathrm{layer}(x) = d_0$*.*
3. *Output* x *as the encoding.*

Lemma 6. *Consider the encoding* $f\colon \mathcal{M}\times\mathcal{R} \to [\mathsf{w}]^v$ *specified in Construction 4. Then,* f *is incomparable. Further,* f *is* ϵ*-secure against target collision resistance, where for any* t*, we have*

$$\epsilon(t) \le t\cdot\left(\frac{1}{|\mathcal{R}|} + 2^{-\lambda}\right).$$

Proof. Incomparability is trivial, as all outputs of the encoding are in the same layer d_0. For target collision resistance, we again apply Lemma 3 with

$$\sum_{x\in[\mathsf{w}]^v} \widehat{\mu}^2(x) = \ell_{d_0}\frac{1}{\ell_{d_0}^2} = \frac{1}{\ell_{d_0}} \le 2^{-\lambda}.$$

□

4.3 Efficient Implementation

Our encodings use functions Ψ. Here, we show how to efficiently implement them.

Remark 6. Due to rounding, our algorithms only result in a function Ψ that yields a distribution close to the desired one. That is, the actual value of $\widehat{\mu}(x)$ that we get differs from the intended one by some ϵ (e.g., $\epsilon \le 1/\mathsf{w}^v$ for TSL). Then, the collision metric increases by at most $\sum_x(2\epsilon\widehat{\mu}(x)+\epsilon^2)$, which is negligible compared to $2^{-\lambda}$ for all instances we consider in Section 5. One can further minimize this gap by increasing the output domain of the random oracle H (and hence the input domain of Ψ). For ease of exposition, we ignore these rounding issues in the following.

Mapping Integers To Vertices and Back. In all three implementations, we will need a way to bijectively map between integers $x \in \{0,\ldots,\ell_d-1\}$ and vertices of the layer $\mathcal{L}_d$. Here, we present algorithms for this. As our algorithms deal with different hypercubes, we make the hypercube more explicit in our notation: we denote the layer d of hypercube $[\mathsf{w}]^v$ as $\mathcal{L}_d^{\langle v\rangle}$ and its size as $\ell_d^{\langle v\rangle}$. The core of the algorithms is that we can relate a layer in a hypercube of dimension v to layers of the hypercube of dimension $v-1$. This is stated in the following two lemmas.

Lemma 7. *Fix a hypercube* $[\mathsf{w}]^v$ *and a vertex* $a=(a_1,a_2,\ldots,a_v) \in [\mathsf{w}]^v$*. Then,* a *belongs to layer* $\mathcal{L}_d^{\langle v\rangle}$ *if and only if* $a_1 \ge \mathsf{w}-d$ *and* $(a_2,\ldots,a_v)$ *belongs to* $\mathcal{L}_{d-(\mathsf{w}-a_1)}^{\langle v-1\rangle}$*, i.e., the layer* $(d-(\mathsf{w}-a_1))$ *of hypercube* $[\mathsf{w}]^{v-1}$*.*

Proof. By definition we have

$$\mathrm{layer}(a) = d \;\Leftrightarrow\; \sum_{1\le i\le v} a_i = v\mathsf{w}-d \;\Leftrightarrow\; \sum_{2\le i\le v} a_i = (v-1)\mathsf{w} - (d-(\mathsf{w}-a_1)).$$

□

Note that for given w, v, d not all a_1 are possible. The set of possible a_1 values is given by the following system of inequations:

$$\begin{cases} 1 \le a_i \le \mathsf{w}; \\ 0 \le d-(\mathsf{w}-a_i) \le (\mathsf{w}-1)(v-1). \end{cases}$$

Lemma 8. *Fix a hypercube* $[\mathsf{w}]^v$ *and a layer* $d \in \{0, \dots, v(\mathsf{w}-1)\}$. *Then, we have*

$$\ell_d^{\langle v \rangle} = \sum_{\max(\mathsf{w}-d,1) \le a_1 \le \min(\mathsf{w},\mathsf{w}+(\mathsf{w}-1)(v-1)-d)} \ell_{d-(\mathsf{w}-a_1)}^{\langle v-1 \rangle}.$$

Proof. By Lemma 7, the cardinality of layer $\mathcal{L}_d^{\langle v \rangle}$ can be rewritten as follows:

$$\begin{aligned}
\ell_d^{\langle v \rangle} &= |\{a \in [\mathsf{w}]^v \mid \mathrm{layer}(a) = d\}| \\
&= \left| \bigcup_{\max(\mathsf{w}-d,1) \le a_1 \le \min(\mathsf{w},\mathsf{w}+(\mathsf{w}-1)(v-1)-d)} \{(a_1, a') \in [\mathsf{w}] \times [\mathsf{w}]^{v-1} \mid \mathrm{layer}(a') = d - (\mathsf{w}-a_1)\} \right| \\
&= \sum_{\max(\mathsf{w}-d,1) \le a_1 \le \min(\mathsf{w},\mathsf{w}+(\mathsf{w}-1)(v-1)-d)} \left| \{(a_1, a') \in [\mathsf{w}] \times [\mathsf{w}]^{v-1} \mid \mathrm{layer}(a') = d - (\mathsf{w}-a_1)\} \right| \\
&= \sum_{\max(\mathsf{w}-d,1) \le a_1 \le \min(\mathsf{w},\mathsf{w}+(\mathsf{w}-1)(v-1)-d)} \ell_{d-(\mathsf{w}-a_1)}^{\langle v-1 \rangle}.
\end{aligned}$$

□

Having these lemmas, we now present the algorithms mapping from integers to vertices in a specific layer of the hypercube and back. We then show that these two algorithms are inverse to each other, and in particular they are bijective.

Construction 5 (Mapping Between Integers and Vertices)**.** *We fix a hypercube* $[\mathsf{w}]^v$ *and layer* d *of it. Consider the following algorithms:*

- $\mathsf{MapToVertex}_{\mathsf{w},v,d}(x) \to a$ *for* $0 \le x < \ell_d^{\langle v \rangle}$ *and* $a \in [\mathsf{w}]^v$
  1. *Set* $x_1 := x$ *and* $d_1 := d$.
  2. *For* i *from* 1 *to* $v-1$*:*
     (a) *Find* $j_i \in [\max(0, d_i - (\mathsf{w}-1)(v-i)); \min(\mathsf{w}-1, d_i)]\}$ *such that (cf. Lemma 8)*
     $$\sum_{\max(0,d_i-(\mathsf{w}-1)(v-i)) \le j < j_i} \ell_{d_i-j}^{\langle v-i \rangle} \le x_i < \sum_{\max(0,d_i-(\mathsf{w}-1)(v-i)) \le j \le j_i} \ell_{d_i-j}^{\langle v-i \rangle}.$$
     (b) *Set* $a_i := \mathsf{w} - j_i \in [\mathsf{w}]$.
     (c) *Set* $d_{i+1} := d_i - j_i = d_i - (\mathsf{w} - a_i)$.
     (d) *Set* $x_{i+1} := x_i - \sum_{\max(0,d_i-(\mathsf{w}-1)(v-1)) \le j < j_i} \ell_{d_i-j}^{\langle v-i \rangle}$.
  3. *Set* $a_v := \mathsf{w} - x_v - d_v$.
  4. *Output* $(a_1, a_2, \dots, a_v)$.
- $\mathsf{MapToInteger}_{\mathsf{w},v,d}(a) \to x$ *for* $0 \le x < \ell_d^{\langle v \rangle}$ *and* $a \in [\mathsf{w}]^v$
  1. *Set* $x_v := 0$ *and* $d_v := \mathsf{w} - a_v$*;*
  2. *For* i *from* $v-1$ *downto* 1*:*
     (a) *Set* $j_i := \mathsf{w} - a_i$.
     (b) *Set* $d_i := d_{i+1} + j_i$.
     (c) *Set* $x_i := x_{i+1} + \sum_{\max(0,d_i-(\mathsf{w}-1)(v-1)) \le j < j_i} \ell_{d_i-j}^{\langle v-i \rangle}$.
  3. *Output* $x := x_1$ *and* $d := d_1$.

Lemma 9. *Fix a hypercube* $[\mathsf{w}]^v$ *and layer* d *of it, and consider the algorithms presented in Construction 5. Then, for any* $0 \le x < \ell_d^{\langle v \rangle}$ *and* $a \in [\mathsf{w}]^v$*, we have*

$$\begin{aligned}
\mathsf{MapToInteger}_{\mathsf{w},v,d}(\mathsf{MapToVertex}_{\mathsf{w},v,d}(x)) &= x, \\
\mathsf{MapToVertex}_{\mathsf{w},v,d}(\mathsf{MapToInteger}_{\mathsf{w},v,d}(a)) &= a.
\end{aligned}$$

In particular, these algorithms induce bijections.

Proof. This can easily be verified. But for completeness, we give a proof by induction on v.

$v = 1$. Here, $\ell_d^{\langle v \rangle} = 1$ for $d \in [0; \mathsf{w} - 1]$. Then MapToVertex maps the only possible $x = 0$ to $a_v := \mathsf{w} - d$. MapToInteger always outputs 0 for $v = 1$.

$v > 1$. Suppose the algorithms are inverses to each other for all $v' < v$. Now, given x and d, the composition of MapToVertex and MapToInteger works as follows:

1. MapToVertex sets $x_1 := x$ and $d_1 := d$.
2. MapToVertex finds j_1 such that
$$\sum_{\max(0, d_1 - (\mathsf{w}-1)(v-1)) \le j < j_1} \ell_{d_1 - j}^{\langle v-1 \rangle} \le x_1 < \sum_{\max(0, d_1 - (\mathsf{w}-1)(v-1)) \le j \le j_1} \ell_{d_1 - j}^{\langle v-1 \rangle}$$
3. MapToVertex sets $a_1 := \mathsf{w} - j_1$, $d_2 := d_1 - j_1$, $x_2 := x_1 - \sum_{\max(0, d_1 - (\mathsf{w}-1)(v-1)) \le j < j_1} \ell_{d_1 - j}^{\langle v-1 \rangle}$.
4. The next steps of MapToVertex are the call to the full MapToVertex with $v := v - 1$, $d := d_2$, $x := x_2$. It outputs $(a_2, \ldots, a_v)$
5. The first steps of MapToInteger are the call to the full MapToInteger with $v := v - 1$, $d := d_2$, and $(a_2, \ldots, a_v)$. By induction hypothesis it is the inverse to the previous step and thus outputs x_2 and d_2.
6. MapToInteger sets $j_1 := \mathsf{w} - a_1$ which matches j_1 in Step 2.
7. MapToInteger sets $d_1 := d_2 + j_1$ which matches d_1 in Step 3.
8. MapToInteger sets $x_1 := x_2 + \sum_{\max(0, d_1 - (\mathsf{w}-1)(v-1)) \le j < j_1} \ell_{d_1 - j}^{\langle v-1 \rangle}$, which matches x_1 in Step 3.
9. MapToInteger outputs x_1 and d_1, which matches x and d in Step 1.

The other direction can be seen analogously. □

Implementation for TLFC. For Construction 2, we require a function Ψ mapping from $\{0, 1, \ldots, \mathsf{w}^v - 1\}$ to the hypercube $[\mathsf{w}^v]$, such that for $z \xleftarrow{\$} \{0, 1, \ldots, \mathsf{w}^v - 1\}$ and any fixed $x \in [\mathsf{w}]^v$, we have $\Pr_z[\Psi(z) = x] = \widehat{\mu}_{\mathrm{layer}(x)}$. We can implement it as follows, given $\{\widehat{\mu}_d\}$:

1. Take as input $x \in \{0, 1, \ldots, \mathsf{w}^v - 1\}$.
2. Find the layer $d \in \{0, \ldots, v(\mathsf{w} - 1)\}$ to which we should map:
$$\mathsf{w}^v \sum_{0 \le j < d} \widehat{\mu}_j \ell_j \le x < \mathsf{w}^v \sum_{0 \le j \le d} \widehat{\mu}_j \ell_j.$$
3. Set $x_d := \left\lfloor (x - \mathsf{w}^v \sum_{0 \le j < d} \widehat{\mu}_j \ell_j) / (\widehat{\mu}_d \mathsf{w}^v) \right\rfloor$.
4. Output $a := \mathsf{MapToVertex}_{\mathsf{w},v,d}(x_d)$ (see Construction 5).

Implementation for TL1C. For Construction 3, we need a function Ψ mapping from $\{0, 1, \ldots, \mathsf{w}^v - 1\}$ to the hypercube $[\mathsf{w}^v]$, such that for $z \xleftarrow{\$} \{0, 1, \ldots, \mathsf{w}^v - 1\}$, we have
$$\forall\, x \in [\mathsf{w}]^v\colon \ \Pr_z[\Psi(z) = x] = \begin{cases} 1/\ell_{[0:d_0]}, & \text{if } \mathrm{layer}(x) \le d_0, \\ 0, & \text{if } \mathrm{layer}(x) > d_0. \end{cases}$$

We can implement it as follows, given d_0:

1. Take as input $x \in \{0, 1, \ldots, \mathsf{w}^v - 1\}$.

2. Find the layer $d \in \{0, \ldots, d_0\}$ to which we should map[10]:

$$\mathsf{w}^v \frac{\ell_{[0:d-1]}}{\ell_{[0:d_0]}} \leq x < \mathsf{w}^v \frac{\ell_{[0:d]}}{\ell_{[0:d_0]}}.$$

3. Set $x_d := \lfloor (x\ell_{[0:d_0]} - \mathsf{w}^v \ell_{[0:d-1]})/\mathsf{w}^v \rfloor$.

4. Output $a := \mathsf{MapToVertex}_{\mathsf{w},v,d}(x_d)$ (see Construction 5).

Implementation for TSL. For Construction 4, we want a function Ψ mapping from $\{0, 1, \ldots, \mathsf{w}^v - 1\}$ to $[\mathsf{w}]^v$, such that for $z \xleftarrow{\$} \{0, 1, \ldots, \mathsf{w}^v - 1\}$, we have

$$\forall\, x \in [\mathsf{w}]^v\colon \ \Pr_z[\Psi(z) = x] = \begin{cases} 1/\ell_{d_0}, & \text{if } \mathrm{layer}(x) = d_0, \\ 0, & \text{if } \mathrm{layer}(x) \neq d_0. \end{cases}$$

We can implement it as follows, given d_0:

1. Take as input $x \in \{0, 1, \ldots, \mathsf{w}^v - 1\}$.

2. Set $x_{d_0} := \lfloor x\ell_{d_0}/\mathsf{w}^v \rfloor$.

3. Output $a := \mathsf{MapToVertex}_{\mathsf{w},v,d}(x_{d_0})$ (see Construction 5).

5 Evaluation

In this section, we compare our encodings to classical encodings and our lower bound.

Encoding Variants. We consider different variants of the one-time signature scheme based on hash chains, using the same security level λ and signature size v (i.e., size measured in the number of chains):

- The regular Winternitz scheme with a checksum [Mer88, Hül13], denoted by WOTS. We select the chain length w and the number v' of base chains so that $\mathsf{w}^{v'} \geq 2^\lambda$ and $v - v'$ chains would suffice for the checksum: $\mathsf{w}^{v-v'} \geq (\mathsf{w} - 1)v' + 1$.
- Winternitz with a constant sum, denoted by WOTS-CS [ZCY23]. Here, the message is mapped into the middle (i.e., largest) layer of hypercube $[\mathsf{w}]^v$ so we select the minimum w such that $\ell^{\langle v \rangle}_{(\mathsf{w}-1)v/2} \geq 2^\lambda$.
- Winternitz with a target sum, denoted by WOTS-TS [DKKW25]. Here, the message repeatedly hashed until the result is in the middle layer of hypercube $[\mathsf{w}]^v$, with $\mathsf{w}^v \geq 2^\lambda$.

We consider various signature sizes from $v = 25$ to $v = 165$ in order to provide an extensive study of the tradeoffs. We try different security levels: $\lambda = 128$ and $\lambda = 160$. In order to find the best parameters for our schemes, we try various chain lengths w from 2 to 96 and select the best variant. For the lower bound, we also try a selection of w, observing that the bound decreases when w grows. However, it quickly stabilizes: for every v there is a w such that bigger chains change the bound negligibly. For our choices of v, the lower bound is stable for all $\mathsf{w} \geq 100$. The results are given in Table 1, whereas the chain lengths for all the instances are given in Table 2.

Script. We provide the script that was used to prepare these tables[11]. In this script, for selected v, w we compute the layer size values $\ell^{\langle v \rangle}_d$ explicitly using the formula [ZCY23, DKKW25]

$$\ell^{\langle v \rangle}_d = \sum_{0 \leq s \leq \lfloor \frac{d}{w} \rfloor} (-1)^s \binom{v}{s} \binom{d - s \cdot w + v - 1}{v - 1}.$$

[10] For convenience, say $\ell_{[0:-1]} = 0$.

[11] See https://github.com/khovratovich/hypercube.

Interpretation. We interpret the results in Table 1 as follows:

- The existing schemes underperform significantly compared to the lower bound given in Theorem 1. For the best of those, WOTS-TS, the ratio between its verification cost and the lower bound ranges between 1.6 (bigger signatures) and 1.2-1.4 (smaller signatures).
- Our new constructions almost match the lower bound for bigger signature sizes, but then the gap appears too. As a result, the best of our schemes improves the verification cost by 60% for 128 chains, by 35% for 64 chains, by 20% for 30 chains. The best result is achieved at chain lengths roughly double the length of Winternitz ones.
- Of the new constructions, the TLFC one is marginally better for big signatures, but suffers from the checksum overhead for small signatures. The other two constructions show identical perfomance, since for our parameters a layer d_0 in $[\mathsf{w}]^v$ (TSL scheme) is essentially equivalent to the set of $d_0 + 1$ top layers in $[\mathsf{w}]^{v-1}$ (part of TL1C), so adding a 1-chain checksum levels the cost.
- One can apply our new encodings to XMSS [BDH11, HBG+18] and we expect the relative performance gains to carry over with almost no change. Namely, the XMSS construction adds a Merkle proof to each one-time signature, so both lower bound and the verification times for our constructions increase by an additive constant, which depends on the tree size (i.e., key lifetime) only. When moving to hyper-tree constructions like SPHINCS$^+$ [HBD+22], the verification cost also scales with the number of tree layers, so the improvement will be roughly similar to plain XMSS. We note, however, that the concrete numbers that we have apply to a single instance of the one-time signature, whereas the security of XMSS and SPHINCS$^+$ relies on the security of multiple instances. Therefore, before using our encodings in these contexts, a careful derivation of concrete parameters is needed.

		Winternitz variants			Lower bound	Our schemes		
	Sig size v	WOTS	WOTS-CS	WOTS-TS		TLFC	TL1C	TSL
Security level $\lambda = 128$	136	67.5	68	68	37.8	38.9	39	39
	132	131.2	66	66	38.2	39.9	39	39
	128	127	128	64	38.9	40	40	40
	86	85.4	86	86	51.3	53	53	53
	84	126.3	84	84	52.1	54.9	54	54
	81	121.9	121	81	53.7	56	56	56
	68	101.5	102	102	62.6	66	65	65
	67	132.3	100	100	63.6	67	66	66
	64	127	128	96	66.6	70.9	70	70
	55	135.6	137	137	78.4	84	83	83
	50	149.6	150	125	88	95.8	93	93
	45	177	157	157	101.4	111.9	108	108
	40	214.5	200	180	120.9	136.7	131	131
	35	257.9	245	210	152.1	177.7	168	168
	30	376.5	330	285	207.5	255.4	235	235
	25	574.8	525	425	324.1	436	384	384
Security level $\lambda = 160$	168	83.5	84	84	47.1	48	48	48
	165	164.6	82	82	47.6	48.9	49	49
	160	159.8	160	80	48.5	49.9	50	50
	106	105.4	106	106	64.3	67	66	66
	104	154.4	104	104	65.3	68	67	67
	101	149.7	151	101	66.9	69	69	69
	84	126.3	126	126	78.8	82.9	82	82
	80	157.6	160	120	82.8	86.9	86	86
	70	174.1	175	140	95.3	100.9	99	99
	60	180	180	180	115.3	123.8	121	121
	50	247.1	225	225	150.2	165	160	160
	45	306	270	247.5	179.5	201.7	193	193
	40	396.5	340	300	224.8	259.7	245	245
	35	526.5	472	402	301.7	363.5	337	337

Table 1: Verification cost of various schemes, expressed in the number of hashes. WOTS – basic Winternitz with a checksum, WOTS-CS – middle layer constant-sum Winternitz, WOTS-TS – target-sum Winternitz with resampling, TLFC, TL1C, TSL – our schemes. Outliers for WOTS, WOTS-CS, WOTS-TS mean that the particular number of chains is suboptimal for the scheme (to attain the security level, the chain must be too long and adds high cost).

		Winternitz variants			Our schemes		
	Sig size v	WOTS	WOTS-CS	WOTS-TS	TLFC	TL1C	TSL
Security level $\lambda = 128$	136	2	2	2	4	4	4
	132	3	2	2	4	5	5
	128	3	3	2	6	4	4
	86	3	3	3	8	6	6
	84	4	3	3	6	6	6
	81	4	4	3	8	6	6
	68	4	4	4	9	9	9
	67	5	4	4	10	9	9
	64	5	5	4	9	8	8
	55	6	6	6	14	10	10
	50	7	7	6	13	14	14
	45	9	8	8	17	18	18
	40	12	11	10	21	20	20
	35	16	15	13	30	26	26
	30	27	23	20	47	44	44
	25	48	43	35	95	86	86
Security level $\lambda = 160$	168	2	2	2	6	4	4
	165	3	2	2	4	4	4
	160	3	3	2	4	4	4
	106	3	3	3	7	6	6
	104	4	3	3	7	7	7
	101	4	4	3	9	6	6
	84	4	4	4	10	7	7
	80	5	5	4	10	8	8
	70	6	6	5	11	13	13
	60	7	7	7	15	14	14
	50	11	10	10	28	20	21
	45	15	13	12	28	28	28
	40	21	18	16	40	44	44
	35	32	28	24	61	56	56

Table 2: Hypercube chain length w for various schemes from Table 1. WOTS– basic Winternitz with a checksum, WOTS-CS – middle layer constant-sum Winternitz, WOTS-TS – target-sum Winternitz with resampling, TLFC, TL1C, TSL – our schemes. These parameters have been used to compute the numbers in Table 1.

References

[BC93] Jurjen N. Bos and David Chaum. Provably unforgeable signatures. In Ernest F. Brickell, editor, *CRYPTO'92*, volume 740 of *LNCS*, pages 1–14. Springer, Berlin, Heidelberg, August 1993. (Cited on Page 3, 5.)

[BDH11] Johannes A. Buchmann, Erik Dahmen, and Andreas Hülsing. XMSS - A practical forward secure signature scheme based on minimal security assumptions. In Bo-Yin Yang, editor, *Post-Quantum Cryptography - 4th International Workshop, PQCrypto 2011*, pages 117–129. Springer, Berlin, Heidelberg, November / December 2011. (Cited on Page 3, 24.)

[BHK+19] Daniel J. Bernstein, Andreas Hülsing, Stefan Kölbl, Ruben Niederhagen, Joost Rijneveld, and Peter Schwabe. The SPHINCS$^+$ signature framework. In Lorenzo Cavallaro, Johannes Kinder, XiaoFeng Wang, and Jonathan Katz, editors, *ACM CCS 2019*, pages 2129–2146. ACM Press, November 2019. (Cited on Page 4, 5.)

[BHRvV21] Joppe W. Bos, Andreas Hülsing, Joost Renes, and Christine van Vredendaal. Rapidly verifiable XMSS signatures. *IACR TCHES*, 2021(1):137–168, 2021. (Cited on Page 5.)

[BM94] Daniel Bleichenbacher and Ueli M. Maurer. Directed acyclic graphs, one-way functions and digital signatures. In Yvo Desmedt, editor, *CRYPTO'94*, volume 839 of *LNCS*, pages 75–82. Springer, Berlin, Heidelberg, August 1994. (Cited on Page 4.)

[BS20] Dan Boneh and Victor Shoup. A graduate course in applied cryptography. *Draft 0.5*, 2020. (Cited on Page 4, 8.)

[CYK16] Jason Paul Cruz, Yoshio Yatani, and Yuichi Kaji. Constant-sum fingerprinting for winternitz one-time signature. In *2016 International Symposium on Information Theory and Its Applications (ISITA)*, pages 703–707. IEEE, 2016. (Cited on Page 3, 5.)

[DKKW25] Justin Drake, Dmitry Khovratovich, Mikhail Kudinov, and Benedikt Wagner. Hash-based multi-signatures for post-quantum ethereum. Cryptology ePrint Archive, Paper 2025/055, 2025. (Cited on Page 3, 5, 6, 8, 23.)

[Gol87] Oded Goldreich. Two remarks concerning the Goldwasser-Micali-Rivest signature scheme. In Andrew M. Odlyzko, editor, *CRYPTO'86*, volume 263 of *LNCS*, pages 104–110. Springer, Berlin, Heidelberg, August 1987. (Cited on Page 3.)

[HBD+22] Andreas Hülsing, Daniel J. Bernstein, Christoph Dobraunig, Maria Eichlseder, Scott Fluhrer, Stefan-Lukas Gazdag, Panos Kampanakis, Stefan Kölbl, Tanja Lange, Martin M. Lauridsen, Florian Mendel, Ruben Niederhagen, Christian Rechberger, Joost Rijneveld, Peter Schwabe, Jean-Philippe Aumasson, Bas Westerbaan, and Ward Beullens. SPHINCS$^+$. Technical report, National Institute of Standards and Technology, 2022. available at `https://csrc.nist.gov/Projects/post-quantum-cryptography/selected-algorithms-2022`. (Cited on Page 3, 5, 24.)

[HBG+18] Andreas Huelsing, Denis Butin, Stefan-Lukas Gazdag, Joost Rijneveld, and Aziz Mohaisen. XMSS: eXtended Merkle Signature Scheme. RFC 8391, May 2018. (Cited on Page 3, 24.)

[HK22] Andreas Hülsing and Mikhail A. Kudinov. Recovering the tight security proof of SPHINCS$^+$. In Shweta Agrawal and Dongdai Lin, editors, *ASIACRYPT 2022, Part IV*, volume 13794 of *LNCS*, pages 3–33. Springer, Cham, December 2022. (Cited on Page 4.)

[HKRY23] Andreas Hülsing, Mikhail A. Kudinov, Eyal Ronen, and Eylon Yogev. SPHINCS+C: Compressing SPHINCS+ with (almost) no cost. In *2023 IEEE Symposium on Security and Privacy*, pages 1435–1453. IEEE Computer Society Press, May 2023. (Cited on Page 3, 5.)

[Hül13] Andreas Hülsing. W-OTS+ - shorter signatures for hash-based signature schemes. In Amr Youssef, Abderrahmane Nitaj, and Aboul Ella Hassanien, editors, *AFRICACRYPT 13*, volume 7918 of *LNCS*, pages 173–188. Springer, Berlin, Heidelberg, June 2013. (Cited on Page 4, 23.)

[KCLM22] Irakliy Khaburzaniya, Konstantinos Chalkias, Kevin Lewi, and Harjasleen Malvai. Aggregating and thresholdizing hash-based signatures using STARKs. In Yuji Suga, Kouichi Sakurai, Xuhua Ding, and Kazue Sako, editors, *ASIACCS 22*, pages 393–407. ACM Press, May / June 2022. (Cited on Page 3, 5.)

[Lam79] Leslie Lamport. Constructing digital signatures from a one-way function. Technical Report SRI-CSL-98, SRI International Computer Science Laboratory, October 1979. (Cited on Page 3.)

[Mer79] Ralph Charles Merkle. *Secrecy, authentication, and public key systems.* Stanford university, 1979. (Cited on Page 3.)

[Mer88] Ralph C. Merkle. A digital signature based on a conventional encryption function. In Carl Pomerance, editor, *CRYPTO'87*, volume 293 of *LNCS*, pages 369–378. Springer, Berlin, Heidelberg, August 1988. (Cited on Page 3, 4, 23.)

[PZC+21] Lucas Pandolfo Perin, Gustavo Zambonin, Ricardo Felipe Custódio, Lucia Moura, and Daniel Panario. Improved constant-sum encodings for hash-based signatures. *Journal of Cryptographic Engineering*, 11(4):329–351, November 2021. (Cited on Page 3, 5.)

[Vau93] Serge Vaudenay. One-time identification with low memory. In *Eurocode'92: International Symposium on Coding Theory and Applications*, pages 217–228. Springer, 1993. (Cited on Page 3, 5.)

[ZCY23] Kaiyi Zhang, Hongrui Cui, and Yu Yu. Revisiting the constant-sum winternitz one-time signature with applications to SPHINCS$^+$ and XMSS. In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part V*, volume 14085 of *LNCS*, pages 455–483. Springer, Cham, August 2023. (Cited on Page 3, 4, 5, 6, 23.)