Short Identity-Based Signatures with Tight Security from Lattices

Jiaxin Pan[1] and Benedikt Wagner*[2]

[1] Department of Mathematical Sciences, NTNU - Norwegian University of Science and Technology, Trondheim, Norway
jiaxin.pan@ntnu.no
[2] KIT - Karlsruhe Institute of Technology, Karlsruhe, Germany
udpto@student.kit.edu

Abstract. We construct a short and adaptively secure identity-based signature scheme *tightly* based on the well-known Short Integer Solution (SIS) assumption. Although identity-based signature schemes can be tightly constructed from either standard signature schemes against adaptive corruptions in the multi-user setting or a two-level hierarchical identity-based encryption scheme, neither of them is known with short signature size and tight security based on the SIS assumption. Here "short" means the signature size is independent of the message length, which is in contrast to the tree-based (tight) signatures.

Our approach consists of two steps: Firstly, we give two generic transformations (one with random oracles and the other without) from non-adaptively secure identity-based signature schemes to adaptively secure ones tightly. Our idea extends the similar transformation for digital signature schemes. Secondly, we construct a non-adaptively secure identity-based signature scheme based on the SIS assumption in the random oracle model.

Keywords. Identity-based signatures, tight security, short integer solution assumption, lattices.

1 Introduction

Tight Security. In public-key cryptography, we often prove the security of a scheme by reductions. Namely, we prove that, if there is an adversary $\mathcal{A}$ that can break the security of a scheme, then we can construct a reduction $\mathcal{R}$ to solve some hard problem (for instance, the short integer solution (SIS) problem [2]). More precisely, by doing this, we establish the relation that $\varepsilon_{\mathcal{A}} \leq \ell \cdot \varepsilon_{\mathcal{R}}$ and the running time of $\mathcal{A}$ and $\mathcal{R}$ are roughly the same, where $\varepsilon_{\mathcal{A}}$ and $\varepsilon_{\mathcal{R}}$ are the success probability of $\mathcal{A}$ and $\mathcal{R}$, respectively.

In particular, if the security loss ℓ is a small constant, then we call the reduction *tight* [5,7]. Recently, a relaxed notion called "almost tight" is considered [12,20], where the security loss can be dependent linearly or logarithmically on the security parameter. A cryptographic scheme with tight reductions does not need to increase the key length to compensate a security loss.

* This work was done while the second author was doing an internship with the first author at NTNU, and it was partially supported by the Erasmus+ traineeship program.

This is the full version of [43], which appeared at PQCrypto 2021.

In the recent years, many tools have been developed to construct tightly secure cryptosystems [29,28,19,20,12,9,35]. However, currently, many of these techniques crucially require pairing groups and the Diffie-Hellman assumption which is known to be insure against a powerful quantum computer. The digital signature scheme in [1,8] and the identity-based encryption schemes in [10,31] are among the few exceptions which have tight post-quantum security using lattice-based techniques.

Our goal: Identity-based signatures with tight post-quantum security. We are interested in advanced cryptographic schemes with tight post-quantum security. In this paper, we consider identity-based signature (IBS) schemes [45]. In an IBS, an honest user with identity id can sign a message m using its secret key $\mathsf{sk}_{\mathsf{id}}$, and a signature σ can be publicly verified, given the master public key mpk and a user's identity id. We are interested in the adaptive security of IBS schemes, where an adversary aims at forging a fresh signature after adaptively learning users' secret keys and signatures. The use of identity-based cryptography can simplify the PKI requirements, and we refer [32] for more discussion about that.

There are mainly two approaches to construct IBS schemes, but neither of them directly yields an IBS scheme with tight post-quantum security. The first approach [16,6] is to transform a (standard) signature scheme into an IBS, which is often referred as the certification approach. The generic transformations in [16,6] are not tight. Recently, it has been shown that, if the underlying signature scheme is tightly secure in the multi-user setting with adaptive corruption, then the IBS scheme is tightly secure [38].

Compared to the classical unforgeability in the single-user setting (EUF-CMA) [26], the multi-user security with corruption (MU-EUF-CMA$^{\mathsf{corr}}$) [3] is a stronger security notion for signature schemes, where an adversary receives verification keys of multiple users and is allowed corrupt some of them. Although EUF-CMA non-tightly implies MU-EUF-CMA$^{\mathsf{corr}}$, constructing a tightly MU-EUF-CMA$^{\mathsf{corr}}$ secure signature scheme is highly challenging: To the best of our knowledge, [24,3,15] are the only schemes that have tight MU-EUF-CMA$^{\mathsf{corr}}$ security, and they are all based on the Diffie-Hellman assumption. There is a generic construction in [3], but it requires a non-interactive witness-indistinguishable proof of knowledge (NIWIPoK) system. It is unclear how to construct this particular proof system and to instantiate their generic construction in the post-quantum setting, while in the pairing setting we have the Groth-Sahai system [27] to implement it.

To give a bit more technical insights to it, in the lattice setting, one may consider to use a proof system (for instance, the one in [14]) together with the OR-proof technique [13] to construct such a NIWIPoK system. However, it requires the rewinding technique to show the PoK property, which leads to a non-tight reduction.

The second approach [32] is to transform a 2-level hierarchical IBE (HIBE) [23] tightly to an IBS scheme. However, the existing tightly secure HIBE schemes are pairing-based [35,36,37]. We note that in [10] Boyen and Li proposed an almost tightly secure lattice-based IBE and claimed (without a concrete scheme)

that it can be turned into a 2-level HIBE. Their construction is motivated by the Katz-Wang "random-bit" technique [25], and it is rather inefficient, due to the use of lattice-based PRFs and their homomorphic evaluation. Lattice-based PRFs [4,30,33] often use a large modulus and have almost tightness only. In Section 1.2, we further sketch why the "random-bit" technique is not enough for achieving our goal.

Our results. We propose the *first* tightly secure IBS scheme based on lattices. We prove the tight adaptive security of our IBS scheme based on the short integer solution (SIS) assumption [2] which is a quantum-safe assumption. Our proof is in the random oracle model. Different to the tree-based tight SIS-based signature scheme in [8], our signatures are short and contain only constant number of elements. Our scheme uses the Micciancio-Peikert (MP) trapdoor technique [41] and the Bonsai tree technique [11]. Moreover, our construction does not require any lattice-based PRF as in [10].

1.1 Technical Details

We achieve our results in two steps.

Step 1: IBS with non-adaptive security. We consider a (weaker) non-adaptive security of IBS schemes, where an adversary has to commit its user secret key queries and signing queries before receiving the master public key. This weaker security gives rise to a tight construction. The main reason is that in the security proof, since adversaries' user secret key queries and signing queries are committed in advance, the reduction can tightly embed the SIS instances in the forgery without any guessing.

More precisely, the overall idea of our non-adaptively secure scheme is as follows. The master public key of our scheme is a random matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ with $m \geq 3n \log q$ where the SIS assumption holds, and the master secret key is a MP trapdoor [41] for $\mathbf{A}$. For generating user secret keys and signing, we associate matrices $\mathbf{F}_{\mathsf{id}} := [\mathbf{A} \mid \mathsf{H}_1(\mathsf{id})] \in \mathbb{Z}_q^{n \times (m+n\lceil \log q \rceil)}$ and $\mathbf{F}_{\mathsf{id},\mathsf{m}} := [\mathbf{A} \mid \mathsf{H}_1(\mathsf{id}) \mid \mathsf{H}_2(\mathsf{id},\mathsf{m})] \in \mathbb{Z}_q^{n \times (m+2n\lceil \log q \rceil)}$ with identity id and message m, where $\mathsf{H}_1, \mathsf{H}_2 : \{0,1\}^* \to \mathbb{Z}_q^{n \times n\lceil \log q \rceil}$ will be simulated as random oracles in the security proof. The secret key of identity id is a MP trapdoor for $\mathbf{F}_{\mathsf{id}}$. Given the trapdoor of $\mathbf{A}$, this can be efficiently computed using trapdoor delegation, e.g. the Bonsai technique [11]. The signature for message m under identity id is a "short" integer vector $\mathbf{z}$ in the kernel of $\mathbf{F}_{\mathsf{id},\mathsf{m}}$ (namely, $\mathbf{F}_{\mathsf{id},\mathsf{m}} \cdot \mathbf{z} = \mathbf{0}$).

Now we are ready to sketch our tight proof. We denote the list of all identities id for user secret key queries as $\mathcal{L}_{id}$, and the list of all identity-message pairs (id,m) for signing queries as $\mathcal{L}_m$. An adversary $\mathcal{A}$ has to output these two lists before receiving the master public key. The key step in our proof is that, by programming the random oracles H_1 and H_2, the reduction can embed a gadget matrix into $\mathbf{F}_{\mathsf{id}}$ (for all $\mathsf{id} \in \mathcal{L}_{id}$) and $\mathbf{F}_{\mathsf{id},\mathsf{m}}$ (for all $(\mathsf{id},\mathsf{m}) \in \mathcal{L}_m$) so that efficiently inverting the SIS function for these $\mathbf{F}_{\mathsf{id}}$ and $\mathbf{F}_{\mathsf{id},\mathsf{m}}$ is possible. However, for all $\mathsf{id}^* \notin \mathcal{L}_{id}$ and $(\mathsf{id}^*,\mathsf{m}^*) \notin \mathcal{L}_m$, $\mathbf{F}_{\mathsf{id}^*}$ and $\mathbf{F}_{\mathsf{id}^*,\mathsf{m}^*}$ are random matrices and inverting

the SIS function for these random matrices is hard. Here, the reduction does not need to guess the forgery $(\mathsf{id}^*, \mathsf{m}^*)$, and thus it is tight.
Step 2: From non-adaptive to adaptive security. For digital signature schemes, it is known that, using a chameleon hash, the non-adaptive security can be tightly transformed to adaptive security [34]. This transformation has been used in the lattice-setting as well [11,41] with the SIS-based chameleon hash function [11]. In this paper, we extend this generic transformation to the IBS setting, and thus our tightly non-adaptively secure IBS yields a tight scheme with adaptive security. Moreover, we propose a more efficient transformation in the random oracle model (cf. Section 3.2), since our non-adaptively secure scheme uses random oracles already. The common practice of instantiating the random oracle with a hash function such as SHA3 will be more efficient than using the chameleon hash technique. In particular, signature sizes are roughly the same, whereas the chameleon hash based on SIS requires to add a matrix to the public key. Further, the computation of this chameleon hash function is less efficient than an highly optimized evaluation of SHA3.
Extension and future work. We note that our approach can be extended to construct hierarchical IBS schemes. We leave this as a future work.

Further, we only analyze security in the (classical) random oracle model. Since our proof does not adaptively program the random oracle, an analysis in the quantum random oracle may be possible. As this model is more desirable for post-quantum cryptography, we think it is an interesting question for future work. We leave constructing an efficient non-adaptively secure short IBS with tight security in the standard model as an open problem. In combination with our transformation from Section 3, this will lead to an adaptively secure short IBS in the standard model with tight security. Finally, our techniques may be transferred to the ring setting, which may be of interest in terms of efficiency.

1.2 More on Related Work

The Katz-Wang "random-bit" technique. The "random-bit" technique can be used to turn the non-tight Gentry-Peikert-Vaikuntanathan (GPV) IBE [22] to a tightly secure one [10,31]. However, we suppose this technique is not useful to construct a tightly secure 2-level HIBE. The high-level idea can be sketched easily. In the tight IBE, the secret key of identity id can be viewed as the GPV secret key of identity $(\mathsf{id}, b_{\mathsf{id}})$, where $b_{\mathsf{id}} \in \{0,1\}$ is a random bit associated with id, and the ciphertext of m under id^* contains two GPV ciphertexts $(\mathsf{c}_0, \mathsf{c}_1)$ of m under identity $(\mathsf{id}^*, 0)$ and $(\mathsf{id}^*, 1)$. In the security proof, by putting a lattice trapdoor in the b_{id}-side and embedding an LWE instance in $(1 - b_{\mathsf{id}})$-side for all identities id, we generate secret keys for identities $(\mathsf{id}, b_{\mathsf{id}})$ and randomize $\mathsf{c}_{1-b_{\mathsf{id}^*}}$ in the challenge ciphertext. This is the key step, and it is important that b_{id^*} to be perfectly hidden, as otherwise the adversary may attack the side without any LWE instance.

One can extend this "random-bit" idea to the 2-level HIBE in the natural manner, namely, the secret key of identity $(\mathsf{id}_1, \mathsf{id}_2)$ is the secret key of identity $(\mathsf{id}_1, b_{\mathsf{id}_1}, \mathsf{id}_2, b_{\mathsf{id}_2})$ in the 2-level GPV HIBE. The encryption algorithm is adapted

accordingly. Imagine that $(\mathsf{id}_1^*, \mathsf{id}_2^*)$ is the challenge identity for the 2-level HIBE. An adversary can learn the bit $b_{\mathsf{id}_1^*}$ used by the reduction, by asking a user secret key of $(\mathsf{id}_1^*, \mathsf{id}_2)$ with $\mathsf{id}_2 \neq \mathsf{id}_2^*$. The similar problem will happen, when we directly use this technique in constructing tightly secure IBS. Thus, we believe the Katz-Wang "random-bit" technique is not useful in constructing tightly secure 2-level HIBE or IBS.

Other tightly secure IBS. We note that in [46] a tightly secure IBS scheme is proposed in a weaker security model, where an adversary cannot ask for the secret key of id if a signature has been asked for id. Moreover, their security relies on the factoring-based and Dlog-based assumptions, while ours is based on the quantum-safe SIS assumption.

Comparison with the Certification Approach. Finally, we compare the efficiency of our scheme with those obtained via the certification approach in Table 1. As we mentioned earlier, a digital signature scheme can be turned into an IBS non-tightly. Here we only focus on instantiating this approach with digital signature schemes based on the plain SIS assumption in the random oracle model, namely, [22,39], for a fair comparison, since our scheme is based on the same assumption. We are optimistic that our scheme can be translated into the more efficient ring setting, and we leave it as our future work.

Scheme	tight	$\lvert\mathsf{mpk}\rvert$	$\lvert\sigma\rvert$
Ours + Section 3.1	✓	$2M$	$(3m + 2n \log q)z$
Ours + Section 3.2	✓	M	$(m + 2n \log q)z + 2\omega(\log \lambda)$
GPV [22] + Cert.	✗	M	$M + 2mz$
Lyu [39] + Cert.	✗	$M + n^2 z$	$M + (n^2 + 2m)z + 2\omega(\log \lambda)$

Table 1. Comparison of our results with identity-based signature schemes obtained by applying the certification approach [32]. We use the chameleon hash function given in [11]. All sizes are in bits, where M denotes the size of an SIS matrix and z the size of an element in $\mathbb{Z}_q$.

We note that the certification approach increases the size of user secret keys and signatures. Namely, a user secret key consists of a secret key, a public key and a signature of the underlying signature scheme, and an identity-based signature contains a public key and two signatures. For schemes in [22,39], their public keys contain a matrix. This is the reason why their signature size (in terms of numbers of elements) is quadratic in n, while ours is linear. For schemes based on structured (and hence stronger) assumptions such as Dilithium [17] and Falcon [18], the certification method will lead to a linear overhead, but it is still not tight.

2 Preliminaries

We use standard notation for sets $\mathbb{N}, \mathbb{P}, \mathbb{R}, \mathbb{Z}, \mathbb{Z}_q$ of natural numbers, primes real numbers, integers and integers modulo $q \in \mathbb{N}$, respectively. By $[n] := \{1, \ldots, n\}$ we denote the set of the first n natural numbers. We denote the security parameter by $\lambda \in \mathbb{N}$. All algorithms will get 1^λ either explicit or implicit as an input. A probabilistic algorithm $\mathcal{A}$ is said to be PPT (probabilistic polynomial time) if its running time can be bounded by a polynomial in its input size. We also make use of standard asymptotic notation for positive functions such as ω and O. A function $\nu : \mathbb{N} \to \mathbb{R}$ is negligible in its input λ if $\nu \in \lambda^{-\omega(1)}$. The term $\mathsf{negl}(\lambda)$ always denotes a negligible function. If a function ν is at least $1 - \mathsf{negl}(\lambda)$, we say that it is overwhelming.

Matrices and vectors are written in bold letters. Vectors should be understood as column vectors. The Euclidean norm of a vector $\mathbf{v}$ is denoted by $\|v\|$, and the spectral norm of a matrix $\mathbf{A}$ is denoted by $s_1(\mathbf{A})$.

If $\mathcal{D}$ is a distribution, we write $x \leftarrow \mathcal{D}$ to state that x is sampled from $\mathcal{D}$. If S is a finite set, the notation $x \xleftarrow{\$} S$ states that x is sampled uniformly random from S. The statistical distance of distributions $\mathcal{D}_1, \mathcal{D}_2$ on the support $\mathcal{X}$ is defined to be $\frac{1}{2} \sum_{x \in \mathcal{X}} |\Pr[\mathcal{D}_1 = x] - \Pr[\mathcal{D}_2 = x]|$. If the statistical distance is negligible in λ, we say the distributions are statistically close. The min-entropy is $H_\infty(\mathcal{D}_1) := -\log(\max_{x \in \mathcal{X}} \Pr[\mathcal{D}_1 = x])$.

If A is an algorithm, the notation $y \leftarrow \mathsf{A}(x)$ means that the variable y is assigned to the output of A on input x. Sometimes we make the randomness used by an algorithm explicit by writing $y = \mathsf{A}(x; r)$ if $r \in \{0,1\}^*$ is $\mathsf{A}'s$ randomness. If we want to state that y is a possible output of Alg on input x, we write $y \in \mathsf{A}(x)$. We use the notation $\mathbf{T}(\mathsf{A})$ for running time. In all code-based security games, numerical values are assumed to be implicitly initialized as 0, sets and lists as $\emptyset$. If $\mathbf{G}$ is a game, we write $\mathbf{G}_{\Pi}^{\mathcal{A}}(1^\lambda) \Rightarrow b$ to state that the game $\mathbf{G}$ outputs the bit $b \in \{0,1\}$ considering the adversary $\mathcal{A}$ and the scheme Π.

Definition 1 (Chameleon Hash Function). *A $\varepsilon_{\mathsf{trap}}$-chameleon hash function (CHF) is a triple of* PPT *algorithms* $\mathsf{CHF} = (\mathsf{CHGen}, \mathsf{CHash}, \mathsf{CHColl})$*, where*

- $\mathsf{CHGen}(1^\lambda)$ *takes as input the security parameter* 1^λ *and outputs the hash key* hk *and the trapdoor* td*. We assume that* hk *implicitly defines a message space* $\mathcal{M}_{\mathsf{hk}}$*, a randomness distribution* $\mathcal{R}_{\mathsf{hk}}$ *and hash value space* $\mathcal{H}_{\mathsf{hk}}$*.*
- $\mathsf{CHash}(\mathsf{hk}, \mathsf{m}; r)$ *takes as input the hash key* hk*, a message* $\mathsf{m} \in \mathcal{M}_{\mathsf{hk}}$ *and randomness* $r \in \mathcal{R}_{\mathsf{hk}}$ *and outputs a hash value* $h \in \mathcal{H}_{\mathsf{hk}}$*.*
- $\mathsf{CHColl}(\mathsf{hk}, \mathsf{td}, \mathsf{m}, r, \hat{\mathsf{m}})$ *takes as input the hash key* hk*, a message* $\mathsf{m} \in \mathcal{M}_{\mathsf{hk}}$*, randomness* $r \in \mathcal{R}_{\mathsf{hk}}$ *and a message* $\hat{\mathsf{m}}$ *and outputs a value* $\hat{r} \in \mathcal{R}_{\mathsf{hk}}$*.*
- *For every* $(\mathsf{hk}, \mathsf{td}) \in \mathsf{CHGen}(1^\lambda), \mathsf{m}, \mathsf{m}' \in \mathcal{M}_{\mathsf{hk}}$ *the following distributions have statistical distance at most* $\varepsilon_{\mathsf{trap}}$*:*

$$\left\{ (r, h) \middle| \begin{array}{c} r \leftarrow \mathcal{R}_{\mathsf{hk}}, \\ h := \mathsf{CHash}(\mathsf{hk}, \mathsf{m}; r) \end{array} \right\} \text{ and } \left\{ (r, h) \middle| \begin{array}{c} r' \leftarrow \mathcal{R}_{\mathsf{hk}}, h := \mathsf{CHash}(\mathsf{hk}, \mathsf{m}'; r'), \\ r \leftarrow \mathsf{CHColl}(\mathsf{hk}, \mathsf{td}, \mathsf{m}', r', \mathsf{m}) \end{array} \right\}.$$

If $\varepsilon_{\mathsf{trap}}$ *is negligible in* λ*, we simply say that* CHF *is a chameleon hash function.*

Definition 2 (Collision Resistant CHF). *Let* $\mathsf{CHF} = (\mathsf{CHGen}, \mathsf{CHash}, \mathsf{CHColl})$ *be a chameleon hash function. We say that* CHF *is collision resistant if for every* PPT *algorithm* $\mathcal{A}$ *the following advantage is negligible in* λ*:*

$$\mathsf{Adv}^{\mathsf{coll}}_{\mathcal{A},\mathsf{CHF}}(\lambda) := \Pr\left[\begin{matrix}\mathsf{CHash}(\mathsf{hk},\mathsf{m};r) = \mathsf{CHash}(\mathsf{hk},\mathsf{m}';r') \\ \wedge (\mathsf{m},r) \neq (\mathsf{m}',r')\end{matrix}\middle|\begin{matrix}(\mathsf{hk},\mathsf{td}) \leftarrow \mathsf{CHGen}(1^\lambda) \\ (\mathsf{m},r,\mathsf{m}',r') \leftarrow \mathcal{A}(\mathsf{hk})\end{matrix}\right].$$

We note that chameleon hash functions based on lattice assumptions are known in the literature [11].

Definition 3 (Identity-Based Signature Scheme). *An Identity-based Signature Scheme (IBS) is defined as a tuple of* PPT *algorithms* $\mathsf{IBS} = (\mathsf{Setup}, \mathsf{KeyExt}, \mathsf{Sig}, \mathsf{Ver})$*, where*

- $\mathsf{Setup}(1^\lambda)$ *takes as input the security parameter* 1^λ *and outputs a master public key* mpk *and a master secret key* msk*. We assume that* mpk *implicitly defines a message space* $\mathcal{M} = \mathcal{M}_{\mathsf{mpk}}$ *and an identity space* $\mathcal{ID} = \mathcal{ID}_{\mathsf{mpk}}$*.*
- $\mathsf{KeyExt}(\mathsf{msk},\mathsf{id})$ *takes as input a master secret key* msk *and an identity* $\mathsf{id} \in \mathcal{ID}$ *and outputs a secret key* $\mathsf{sk}_{\mathsf{id}}$*, we assume that* $\mathsf{sk}_{\mathsf{id}}$ *implicitly contains* id*.*
- $\mathsf{Sig}(\mathsf{sk}_{\mathsf{id}},\mathsf{m})$ *takes as input a secret key* $\mathsf{sk}_{\mathsf{id}}$ *and a message* $\mathsf{m} \in \mathcal{M}$ *and outputs a signature* σ*.*
- $\mathsf{Ver}(\mathsf{mpk},\mathsf{id},\mathsf{m},\sigma)$ *is deterministic, takes as input a master public key* mpk*, identity* $\mathsf{id} \in \mathcal{ID}$*, message* $\mathsf{m} \in \mathcal{M}$ *and signature* σ *and outputs a bit* $b \in \{0,1\}$*.*

We say that IBS *is* ρ*-complete, if for every* $(\mathsf{mpk},\mathsf{msk}) \in \mathsf{Setup}(1^\lambda), \mathsf{m} \in \mathcal{M}, \mathsf{id} \in \mathcal{ID}$ *we have*

$$\Pr\left[\mathsf{Ver}(\mathsf{mpk},\mathsf{id},\mathsf{m},\sigma) = 1 \mid \mathsf{sk}_{\mathsf{id}} \leftarrow \mathsf{KeyExt}(\mathsf{msk},\mathsf{id}), \sigma \leftarrow \mathsf{Sig}(\mathsf{sk}_{\mathsf{id}},\mathsf{m})\right] \geq \rho.$$

Definition 4 (Security of IBS). *Let* $\mathsf{IBS} = (\mathsf{Setup},\mathsf{KeyExt},\mathsf{Sig},\mathsf{Ver})$ *be an IBS and consider games* $\mathbf{UF\text{-}CMA}, \mathbf{UF\text{-}naCMA}$ *given in Fig. 1. We say that* IBS *is* $\mathsf{UF\text{-}naCMA}$ *secure, if for every* PPT *adversary* $\mathcal{A}$ *the following advantage is negligible in* λ*:*

$$\mathsf{Adv}^{\mathsf{UF\text{-}naCMA}}_{\mathcal{A},\mathsf{IBS}}(\lambda) := \Pr\left[\mathbf{UF\text{-}naCMA}^{\mathcal{A}}_{\mathsf{IBS}}(\lambda) \Rightarrow 1\right].$$

We say that IBS *is* $\mathsf{UF\text{-}CMA}$ *secure, if for every* PPT *adversary* $\mathcal{A}$ *the following advantage is negligible in* λ*:*

$$\mathsf{Adv}^{\mathsf{UF\text{-}CMA}}_{\mathcal{A},\mathsf{IBS}}(\lambda) := \Pr\left[\mathbf{UF\text{-}CMA}^{\mathcal{A}}_{\mathsf{IBS}}(\lambda) \Rightarrow 1\right].$$

Background on Lattices and Gaussians. For any m-dimensional lattice (i.e. discrete additive subgroup of $\mathbb{R}^m$) Λ and vector $\mathbf{c} \in \mathbb{R}^m$ we denote the discrete Gaussian distribution with parameter $s > 0$ over the coset $\mathbf{c} + \Lambda$ by $D_{\mathbf{c}+\Lambda,s}$. More precisely, this is the distribution proportional to $\rho_s(\mathbf{x}) := \exp(-\pi\|\mathbf{x}\|^2/s^2)$

Game UF-naCMA$^{\mathcal{A}}_{\mathsf{IBS}}(\lambda)$

$(\mathcal{L}_{id}, \mathcal{L}_m, St) \leftarrow \mathcal{A}(1^\lambda)$
$(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$
for $\mathsf{id} \in \mathcal{L}_{id}$:
  $\mathsf{sk}_{\mathsf{id}} \leftarrow \mathsf{KeyExt}(\mathsf{msk}, \mathsf{id})$
  $\mathcal{L}_{sk} := \mathcal{L}_{sk} \cup \{\mathsf{sk}_{\mathsf{id}}\}$
for $(\mathsf{id}, \mathsf{m}) \in \mathcal{L}_m$:
  $\sigma \leftarrow \mathsf{Sig}(\mathsf{sk}_{\mathsf{id}}, \mathsf{m})$
  $\mathcal{L}_{sig} := \mathcal{L}_{sig} \cup \{\sigma\}$
$(\mathsf{id}^*, \mathsf{m}^*, \sigma^*) \leftarrow \mathcal{A}^{\mathsf{H}}(St, \mathsf{mpk}, \mathcal{L}_{sk}, \mathcal{L}_{sig})$
if $\mathsf{id}^* \in \mathcal{L}_{id}$: **return** 0
if $(\mathsf{id}^*, \mathsf{m}^*) \in \mathcal{L}_m$: **return** 0
return $\mathsf{Ver}(\mathsf{mpk}, \mathsf{id}^*, \mathsf{m}^*, \sigma^*)$

Game UF-CMA$^{\mathcal{A}}_{\mathsf{IBS}}(\lambda)$

$(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$
$(\mathsf{id}^*, \mathsf{m}^*, \sigma^*) \leftarrow \mathcal{A}^{\textsc{Key},\textsc{Sig},\mathsf{H}}(1^\lambda, \mathsf{mpk})$
if $\mathsf{id}^* \in \mathcal{L}_{id}$: **return** 0
if $(\mathsf{id}^*, \mathsf{m}^*) \in \mathcal{L}_m$: **return** 0
return $\mathsf{Ver}(\mathsf{mpk}, \mathsf{id}^*, \mathsf{m}^*, \sigma^*)$

Oracle $\textsc{Key}(\mathsf{id})$
$\mathcal{L}_{id} := \mathcal{L}_{id} \cup \{\mathsf{id}\}$
return $\mathsf{KeyExt}(\mathsf{msk}, \mathsf{id})$

Oracle $\textsc{Sig}(\mathsf{id}, \mathsf{m})$
$\mathcal{L}_m := \mathcal{L}_m \cup \{(\mathsf{id}, \mathsf{m})\}$
$\mathsf{sk}_{\mathsf{id}} \leftarrow \mathsf{KeyExt}(\mathsf{msk}, \mathsf{id})$
return $\sigma \leftarrow \mathsf{Sig}(\mathsf{sk}_{\mathsf{id}}, \mathsf{m})$

Fig. 1. The games **UF-naCMA** (left) and **UF-CMA** (right) for an identity-based signature scheme IBS and a random oracle H (in the standard model case, the oracle H is removed).

restricted to the coset $\mathbf{c} + \Lambda$. Let $\mathbf{A} \in \mathbb{Z}_q^{n \times m}, m > n$ be a matrix. It defines an m-dimensional q-ary lattice and lattice cosets as follows:

$$\Lambda_q^\perp(\mathbf{A}) := \{\mathbf{z} \in \mathbb{Z}^m : \mathbf{Az} = \mathbf{0} \mod q\},\ \Lambda_{\mathbf{u}}^\perp(\mathbf{A}) := \{\mathbf{z} \in \mathbb{Z}^m : \mathbf{Az} = \mathbf{u} \mod q\}.$$

We recall some standard facts about these lattices. For simplicity, throughout the paper we just deal with a prime modululs q. However, the techniques can be generalized to composite q as well [41]. The following lemma is obtained from Lemma 2.9 in [40] by setting $t = \sqrt{m} + \sqrt{n} \in \omega(\sqrt{\log m})$ and hence doubling the constant.

Lemma 1. *There is some univeral constant $C_0 > 0$ such that the following holds: Let $n, m \in \mathbb{N}$ and $\mathbf{X} \in \mathbb{R}^{n \times m}$ be a random subgaussian matrix wth parameter s. Then we have $s_1(\mathbf{X}) \leq C_0 \cdot s \cdot (\sqrt{m} + \sqrt{n})$ except with negligible probability.*

The following facts are from [2,42,44,21,22] and can be obtained by using Lemmas 5.1, 5.2 and 5.3 in [21] and Lemma 4.4 in [42].

Lemma 2. *Let $n, m \in \mathbb{N}$, $q \in \mathbb{P}$ at least polynomial in n, $m \geq 2n \log q$. Consider any $\omega(\sqrt{\log m})$ function and $s \geq \omega(\sqrt{\log m})$. Then for all but a negligible (in n) fraction of all $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ the following distribution is statistically close to uniform over $\mathbb{Z}_q^n$: $\{\mathbf{Ae} \mid \mathbf{e} \leftarrow D_{\mathbb{Z}^m, s}\}$. Furthermore, the conditional distribution of $\mathbf{e} \leftarrow D_{\mathbb{Z}^m, s}$ given $\mathbf{u} = \mathbf{Ae} \mod q$ is exactly $D_{\Lambda_{\mathbf{u}}^\perp(\mathbf{A}), s}$.*

Lemma 3. *Let $n \in \mathbb{N}$, $q \in \mathbb{P}$ and $m \geq 2n \log q$. Consider any $\omega(\sqrt{\log m})$ function and $s \geq \omega(\sqrt{\log m})$. Then for all but an at most q^{-n} fraction of all $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and any vector $\mathbf{u} \in \mathbb{Z}_q^n$, we have $\Pr\left[\|\mathbf{x}\| > s\sqrt{m} \mid \mathbf{x} \leftarrow D_{\Lambda_{\mathbf{u}}^\perp(\mathbf{A}), s}\right] \leq 2^{-m+1}$.*

Lemma 4. *Let $n \in \mathbb{N}$, $q \in \mathbb{P}$ and $m \geq 2n\log q$. Consider any $\omega(\sqrt{\log m})$ function and $s \geq \omega(\sqrt{\log m})$. Then for all but an at most q^{-n} fraction of all $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and any vector $\mathbf{u} \in \mathbb{Z}_q^n$, we have $H_\infty\left(D_{\Lambda_{\mathbf{u}}^\perp(\mathbf{A}),s}\right) \geq m-1$.*

Throughout this paper, we let $\mathbf{G}$ be the fixed gadget matrix introduced in [41]. Let $n, m, q \in \mathbb{N}, m \geq n\lceil\log q\rceil$ and $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ be a matrix. A trapdoor for $\mathbf{A}$ is a matrix $\mathbf{R} \in \mathbb{Z}^{(m-n\lceil\log q\rceil)\times n\lceil\log q\rceil}$ such that $\mathbf{A}[-\mathbf{R}^t \mid \mathbf{I}_{n\lceil\log q\rceil}]^t = \mathbf{G}$. The next lemma summarizes the results in [41]. In particular, we obtained the precise statements from Section 5 of [41], we use the statistical instantiation of trapdoors and the constant is $C_1 = \sqrt{s_1(\Sigma_{\mathbf{G}})+2} \leq 3$ (see [41]).

Lemma 5. *Let C_0 be the constant from Lem. 1. There are* PPT *algorithms* $\mathsf{GenTrap}, \mathsf{SampleD}$ *and* $\mathsf{DelTrap}$ *and constants $C_1 \leq 3$ such that for $n, q, m \in \mathbb{N}, q \geq 2, m \geq 3n\log q, w := n\lceil\log q\rceil$ and any $\omega(\sqrt{\log n})$ function the following holds with overwhelming probability over all random choices:*

- *For any $s \geq \omega(\sqrt{\log n})$ the algorithm $\mathsf{GenTrap}(1^n, 1^m, s, q)$ outputs matrices $\mathbf{A} \in \mathbb{Z}_q^{n\times m}, \mathbf{R} \in \mathbb{Z}^{(m-w)\times w}$ such that $\mathbf{A}$ is statistically close to uniform, $\mathbf{R}$ is a trapdoor for $\mathbf{A}$ with entries sampled from $D_{\mathbb{Z},s}$ and $s_1(\mathbf{R}) \leq s \cdot C_0 \cdot (\sqrt{m-w}+\sqrt{w})$.*
- *For any matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ with trapdoor $\mathbf{R}$, for any $\mathbf{u} \in \mathbb{Z}_q^n$ and any $s \geq C_1 \cdot \sqrt{s_1(\mathbf{R})^2+1} \cdot \omega(\sqrt{\log n})$, the following distribution is statistically close to $D_{\Lambda_{\mathbf{u}}^\perp(\mathbf{A}),s}$:*
$$\{\mathbf{z} | \mathbf{z} \leftarrow \mathsf{SampleD}(\mathbf{A}, \mathbf{R}, \mathbf{u}, s)\}.$$
- *For any matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ with trapdoor $\mathbf{R}$, any matrix $\mathbf{A}' \in \mathbb{Z}_q^{n\times w}$ and any $s \geq C_1 \cdot \sqrt{s_1(\mathbf{R})^2+1} \cdot \omega(\sqrt{\log n})$, $\mathsf{DelTrap}([\mathbf{A} \mid \mathbf{A}'], \mathbf{R}, s)$ outputs a trapdoor $\mathbf{R}' \in \mathbb{Z}_q^{m\times w}$ for $[\mathbf{A} \mid \mathbf{A}']$ with distribution independent of $\mathbf{R}$ and $s_1(\mathbf{R}') \leq s \cdot C_0 \cdot (\sqrt{m}+\sqrt{w})$. Further, for $\tilde{s} \geq \omega(\sqrt{\log n})$ and under the same conditions, the following distributions are statistically close:*
$$\left\{(\mathbf{A}, \mathbf{A}', \mathbf{R}') \middle| \begin{array}{c}(\mathbf{A}, \mathbf{R}) \leftarrow \mathsf{GenTrap}(1^n, 1^m, \tilde{s}, q), \mathbf{A}' \xleftarrow{\$} \mathbb{Z}_q^{n\times w}, \\ \mathbf{R}' \leftarrow \mathsf{DelTrap}([\mathbf{A} \mid \mathbf{A}'], \mathbf{R}, s)\end{array}\right\}$$
and
$$\left\{(\mathbf{A}, \mathbf{A}', \mathbf{R}') \middle| (\mathbf{A}, \mathbf{R}) \leftarrow \mathsf{GenTrap}(1^n, 1^m, \tilde{s}, q), \mathbf{R}' \leftarrow D_{\mathbb{Z},s}^{m\times w}, \mathbf{A}' := \mathbf{A}\mathbf{R}' + \mathbf{G}\right\}.$$

Definition 5 (Short Integer Solution Assumption (SIS)). *Let $\lambda \in \mathbb{N}, n = n(\lambda), m = m(\lambda), \beta = \beta(\lambda) \in \mathbb{N}$ and $q = q(n)$ be prime number. We say that the $\mathsf{SIS}_{n,m,q,\beta}$ assumption holds, if for every* PPT *algorithm $\mathcal{A}$ the following advantage is negligible in λ:*
$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{SIS}_{n,m,q,\beta}}(\lambda) := \Pr\left[\mathbf{A}\mathbf{z} = \mathbf{0} \wedge \mathbf{z} \neq \mathbf{0} \wedge \|\mathbf{z}\| \leq \beta \mid \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m}, \mathbf{z} \leftarrow \mathcal{A}(\mathbf{A})\right].$$

The hardness of SIS for certain parameters is supported by several worst-case to average-case reductions, see [2,42,22].

3 Generic Constructions of Adaptively Secure IBS

In this section we will provide two transformations from non-adaptively secure identity-based signature schemes to adaptively secure ones. Let $\mathsf{IBS}' = (\mathsf{Setup}', \mathsf{KeyExt}', \mathsf{Sig}', \mathsf{Ver}')$ be an identity-based signature scheme with identity space $\mathcal{ID}$ and message space $\mathcal{M}$, $\mathsf{CHF} = (\mathsf{CHGen}, \mathsf{CHash}, \mathsf{CHColl})$ a chameleon hash function, $\mathsf{H}_1 : \{0,1\}^* \to \mathcal{ID}, \mathsf{H}_2 : \{0,1\}^* \to \mathcal{M}$ random oracles and $\ell = \ell(\lambda) \in \mathbb{N}$. We define new identity-based signature schemes IBS in Fig. 2 and $\mathsf{IBS}_{\mathsf{ROM}}$ in Fig. 5. Note that the first transformation works in the standard model, whereas the second one uses random oracles. The reason why we also introduce this second transformation is that our non-adaptively secure construction from lattices uses random oracles already, so we can use the second transformation which is more efficient and without relying on additional primitives such as chameleon hash functions.

3.1 Transformation in the Standard Model

Here we will show that if IBS' is non-adaptively secure and CHF is a collision-resistant chameleon hash, then IBS, defined in Fig. 2, is adaptively secure. It is clear that if IBS' is ρ-complete, then IBS is ρ-complete as well.

Alg $\mathsf{Setup}(1^\lambda)$
01 $(\mathsf{hk}, \mathsf{td}) \leftarrow \mathsf{CHGen}(1^\lambda)$
02 $(\mathsf{mpk}', \mathsf{msk}') \leftarrow \mathsf{Setup}'(1^\lambda)$
03 $\mathsf{mpk} := (\mathsf{mpk}', \mathsf{hk}), \mathsf{msk} := \mathsf{msk}'$
04 **return** $(\mathsf{mpk}, \mathsf{msk})$

Alg $\mathsf{KeyExt}(\mathsf{msk}, \mathsf{id})$
05 $r \leftarrow \mathcal{R}_{\mathsf{hk}}$
06 $\mathsf{id}' := \mathsf{CHash}(\mathsf{hk}, \mathsf{id}; r)$
07 $\mathsf{sk}'_{\mathsf{id}} \leftarrow \mathsf{KeyExt}'(\mathsf{msk}', \mathsf{id}')$
08 **return** $(r, \mathsf{sk}'_{\mathsf{id}})$

Alg $\mathsf{Sig}(\mathsf{sk}_{\mathsf{id}}, \mathsf{m})$
09 **let** $\mathsf{sk}_{\mathsf{id}} = (r, \mathsf{sk}'_{\mathsf{id}})$
10 $s \leftarrow \mathcal{R}_{\mathsf{hk}}$
11 $\sigma' \leftarrow \mathsf{Sig}'(\mathsf{sk}'_{\mathsf{id}}, \mathsf{CHash}(\mathsf{hk}, \mathsf{m}; s))$
12 **return** (r, s, σ')

Alg $\mathsf{Ver}(\mathsf{mpk}, \mathsf{id}, \mathsf{m}, \sigma)$
13 **let** $\sigma = (r, s, \sigma')$
14 $\mathsf{id}' := \mathsf{CHash}(\mathsf{hk}, \mathsf{id}; r)$
15 $\mathsf{m}' := \mathsf{CHash}(\mathsf{hk}, \mathsf{m}; s)$
16 **return** $\mathsf{Ver}'(\mathsf{mpk}', \mathsf{id}', \mathsf{m}', \sigma')$

Fig. 2. Our adaptively secure $\mathsf{IBS} = (\mathsf{Setup}, \mathsf{KeyExt}, \mathsf{Sig}, \mathsf{Ver})$ for a given non-adaptively secure $\mathsf{IBS}' = (\mathsf{Setup}', \mathsf{KeyExt}', \mathsf{Sig}', \mathsf{Ver}')$ and a chameleon hash function $\mathsf{CHF} = (\mathsf{CHGen}, \mathsf{CHash}, \mathsf{CHColl})$.

Theorem 1. *Let* IBS' *be an identity-based signature scheme and* CHF *be an* $\varepsilon_{\mathsf{trap}}$*-chameleon hash function. If* IBS' *is* $\mathsf{UF\text{-}naCMA}$ *secure and* CHF *is collision resistant, then* IBS *is* $\mathsf{UF\text{-}CMA}$ *secure. In particular, for every algorithm* $\mathcal{A}$ *making at most* Q_S *signing queries and* Q_C *secret key queries there are algorithms* $\mathcal{B}_1$ *and* $\mathcal{B}_2$ *such that*

$$\mathsf{Adv}^{\mathsf{UF\text{-}CMA}}_{\mathcal{A},\mathsf{IBS}}(\lambda) \leq \mathsf{Adv}^{\mathsf{coll}}_{\mathcal{B}_1,\mathsf{CHF}}(\lambda) + \mathsf{Adv}^{\mathsf{UF\text{-}naCMA}}_{\mathcal{B}_2,\mathsf{IBS}'}(\lambda) + (Q_C + 2Q_S)\varepsilon_{\mathsf{trap}}.$$

and $\mathbf{T}(\mathcal{B}_1) \approx \mathbf{T}(\mathcal{A}), \mathbf{T}(\mathcal{B}_2) \approx \mathbf{T}(\mathcal{A})$.

Proof. We prove the statement via a games $\mathbf{G}_0$ and $\mathbf{G}_1$ and reductions $\mathcal{B}_1$ and $\mathcal{B}_2$. Games $\mathbf{G}_0, \mathbf{G}_1$ are formally presented in Fig. 3. In every game $i \in \{0,1\}$, we denote the advantage of adversary $\mathcal{A}$ as $\mathsf{Adv}_i(\mathcal{A}) := \Pr\left[\mathbf{G}_i^{\mathcal{A}} \Rightarrow 1\right]$. Game $\mathbf{G}_0$ is the original game **UF-CMA**, hence we aim to bound $\mathsf{Adv}_0(\mathcal{A})$. Game $\mathbf{G}_1$

Game $\mathbf{G}_0$,$\mathbf{G}_1$

$(\mathsf{hk}, \mathsf{td}) \leftarrow \mathsf{CHGen}(1^\lambda)$
$(\mathsf{mpk}', \mathsf{msk}') \leftarrow \mathsf{Setup}'(1^\lambda)$
$\mathsf{mpk} := (\mathsf{mpk}', \mathsf{hk}), \mathsf{msk} := \mathsf{msk}'$
$(\mathsf{id}^*, \mathsf{m}^*, (r^*, s^*, \sigma^*)) \leftarrow \mathcal{A}^{\textsc{Key},\textsc{Sig}}(\mathsf{mpk})$
if $\mathsf{id}^* \in \mathcal{L}_{id}$: **return** 0
if $(\mathsf{id}^*, \mathsf{m}^*) \in \mathcal{L}_m$: **return** 0
$\mathsf{id}_h^* := \mathsf{CHash}(\mathsf{hk}, \mathsf{id}^*; r^*)$
$\mathsf{m}_h^* := \mathsf{CHash}(\mathsf{hk}, \mathsf{m}^*; s^*)$
if $\exists(\mathsf{id}', \mathsf{id}, r) \in \mathcal{H}_{id}$: $\mathsf{id}_h^* = \mathsf{id}'$:
  $\mathsf{bad}_1 = 1$, **return** 0
if $\exists((\mathsf{id}', \mathsf{id}, r), (\mathsf{m}', \mathsf{m}, s)) \in \mathcal{H}_m$:
$\mathsf{id}_h^* = \mathsf{id}' \wedge \mathsf{m}_h^* = \mathsf{m}'$:
  $\mathsf{bad}_2 = 1$, **return** 0
return $\mathsf{Ver}(\mathsf{mpk}, \mathsf{id}^*, \mathsf{m}^*, \sigma^*)$

Oracle $\textsc{Key}(\mathsf{id})$

$\mathcal{L}_{id} := \mathcal{L}_{id} \cup \{\mathsf{id}\}$
$r \leftarrow \mathcal{R}_{\mathsf{hk}}, \mathsf{id}' := \mathsf{CHash}(\mathsf{hk}, \mathsf{id}; r)$
$\mathsf{sk}'_{\mathsf{id}} \leftarrow \mathsf{KeyExt}'(\mathsf{msk}', \mathsf{id}')$
$x := (\mathsf{id}', \mathsf{id}, r)$
$\mathcal{H}_{id} := \mathcal{H}_{id} \cup \{x\}$
return $(r, \mathsf{sk}'_{\mathsf{id}})$

Oracle $\textsc{Sig}(\mathsf{id}, \mathsf{m})$

$\mathcal{L}_m := \mathcal{L}_m \cup \{(\mathsf{id}, \mathsf{m})\}$
$r \leftarrow \mathcal{R}_{\mathsf{hk}}, \mathsf{id}' := \mathsf{CHash}(\mathsf{hk}, \mathsf{id}; r)$
$\mathsf{sk}'_{\mathsf{id}} \leftarrow \mathsf{KeyExt}'(\mathsf{msk}', \mathsf{id}')$
$s \leftarrow \mathcal{R}_{\mathsf{hk}}, \mathsf{m}' := \mathsf{CHash}(\mathsf{hk}, \mathsf{m}; s)$
$\sigma' \leftarrow \mathsf{Sig}'(\mathsf{sk}'_{\mathsf{id}}, \mathsf{m}')$
$x := ((\mathsf{id}', \mathsf{id}, r), (\mathsf{m}', \mathsf{m}, s))$
$\mathcal{H}_m := \mathcal{H}_m \cup \{x\}$
return (r, s, σ')

Fig. 3. Games $\mathbf{G}_0$ and $\mathbf{G}_1$ in the proof of Thm. 1. The shaded statements are only executed in $\mathbf{G}_1$.

keeps track of the hashed identities and messages for all key and signing queries. That is, it holds lists $\mathcal{H}_m$ and $\mathcal{H}_{id}$ such that $(\mathsf{id}', \mathsf{id}, r) \in \mathcal{H}_{id}$ means that the adversary asked for a secret key for identity id and when answering the query, the game hashed id to id' using r, i.e. $\mathsf{id}' = \mathsf{CHash}(\mathsf{hk}, \mathsf{id}; r)$. In a similar way, a tuple $((\mathsf{id}', \mathsf{id}, r), (\mathsf{m}', \mathsf{m}, s))$ is in $\mathcal{H}_m$ if in some signing query, id was hashed to id and m was hashed to m' using randomness r, s respectively. After obtaining $\mathcal{A}$'s forgery $(\mathsf{id}^*, \mathsf{m}^*, (r^*, s^*, \sigma^*))$, the game checks additional conditions and returns 0 if one of them holds. These are modeled as the events $\mathsf{bad}_1, \mathsf{bad}_2$. Setting $\mathsf{bad} := \mathsf{bad}_1 \vee \mathsf{bad}_2$ we can bound the difference of both games by

$$|\mathsf{Adv}_0(\mathcal{A}) - \mathsf{Adv}_1(\mathcal{A})| \leq \Pr[\mathsf{bad}].$$

Next, consider the definition of bad_1:

$$\mathsf{bad}_1 := (\exists(\mathsf{id}', \mathsf{id}, r) \in \mathcal{H}_{id} : \mathsf{CHash}(\mathsf{hk}, \mathsf{id}^*; r^*) = \mathsf{id}'),$$

which implies that there is a collision

$$\mathsf{CHash}(\mathsf{hk}, \mathsf{id}^*; r^*) = \mathsf{id}' = \mathsf{CHash}(\mathsf{hk}, \mathsf{id}; r).$$

This collision is non-trivial, as $\mathsf{id}^* \notin \mathcal{L}_{id}$ and hence $\mathsf{id}^* \neq \mathsf{id}$. Similarly, a non-trivial collision can be found if bad_2 occurs. Simulating $\mathbf{G}_1$ and checking which collision

occurs can be done efficiently without the knowledge of the hash trapdoor td, hence we have a direct reduction $\mathcal{B}_1$ that finds a collision for given hk if bad holds true. Clearly the running time of $\mathcal{B}_1$ is dominated by running $\mathcal{A}$ once. We see that

$$\Pr[\mathsf{bad}] \leq \mathsf{Adv}^{\mathsf{coll}}_{\mathcal{B}_1,\mathsf{CHF}}(\lambda).$$

Finally, we bound the advantage of $\mathcal{A}$ in game $\mathbf{G}_1$ by a reduction $\mathcal{B}_2$ playing

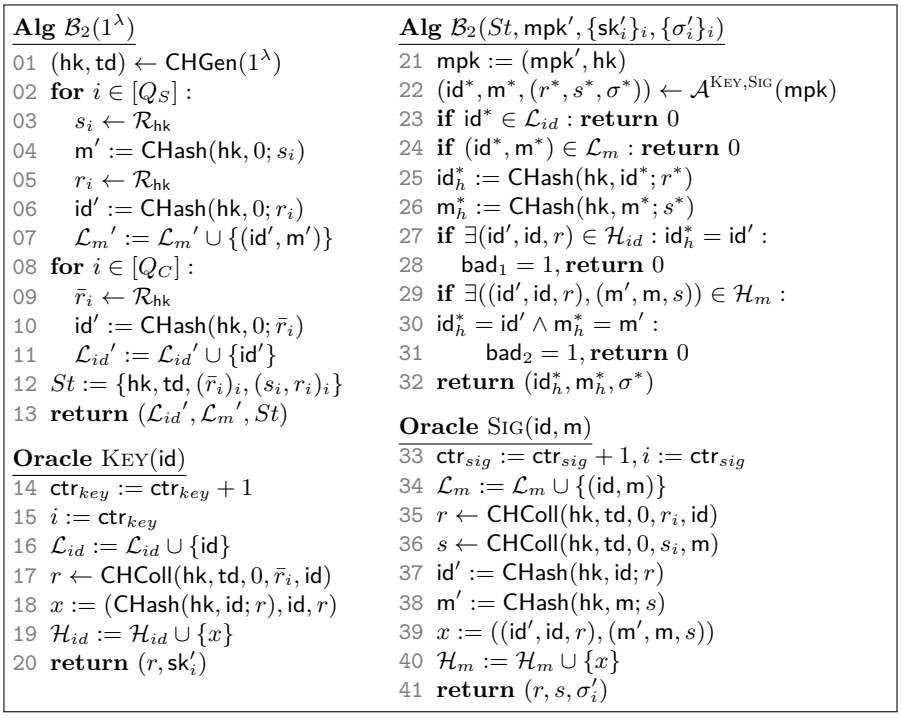

Fig. 4. Reduction $\mathcal{B}_2$ in the proof of Thm. 1 simulating game $\mathbf{G}_1$ for adversary $\mathcal{A}$ and playing the game **UF-naCMA** for the scheme IBS'.

the game **UF-naCMA** for the scheme IBS'. The reduction makes use of the trapdoor td and is formally presented in Fig. 4. Reduction $\mathcal{B}_2$ chooses a chameleon hash key and a trapdoor for it and then hashes Q_S many arbitrary values (in our presentation: 0) using randomness r_i, s_i to hash values $\mathsf{id}'_i, \mathsf{m}'_i$. These hash values will then be given to the UF-naCMA challenger as the (non-adaptive) signing queries. $\mathcal{B}_2$ will then get a public key mpk' and the signatures σ'_i for these queries. Afterwards, when $\mathcal{A}$ issues the i-th (adaptive) signature query for the pair $(\mathsf{id}, \mathsf{m})$, the reduction uses its trapdoor to find randomness r and s, such that $\mathsf{CHash}(\mathsf{hk}, \mathsf{id}; r) = \mathsf{id}'_i$ and $\mathsf{CHash}(\mathsf{hk}, \mathsf{m}; s) = \mathsf{m}'_i$, i.e.

$$r \leftarrow \mathsf{CHColl}(\mathsf{hk}, \mathsf{td}, 0, r_i, \mathsf{id}), s \leftarrow \mathsf{CHColl}(\mathsf{hk}, \mathsf{td}, 0, s_i, \mathsf{m}).$$

Then the reduction can simply return (r, s, σ'_i), which is correct, by definition of the scheme. A similar collision strategy is applied to handle the adaptive secret key queries after non-adaptively obtaining secret keys. After obtaining $\mathcal{A}$'s forgery $(\mathsf{id}^*, \mathsf{m}^*, (r^*, s^*, \sigma^*))$, $\mathcal{B}_2$ checks all the winning conditions and outputs $(\mathsf{CHash}(\mathsf{hk}, \mathsf{id}^*; r^*), \mathsf{CHash}(\mathsf{hk}, \mathsf{m}^*; s^*), \sigma^*)$. It follows from the properties of the chameleon hash function that this collision finding using the trapdoor and honest signing are statistically close. To be precise, the statistical distance between $\mathbf{G}_1$ and the game simulated by $\mathcal{B}_2$ can be bounded by $(Q_C + 2Q_S)\varepsilon_{\mathsf{trap}}$, as $\mathcal{B}_2$ applies CHColl once per secret key query and twice per signing query. Let us now argue that $\mathcal{B}_2$ wins the game **UF-naCMA**, assuming $\mathcal{A}$ wins. By definition of the verification algorithm of the scheme, if $\mathcal{A}$ outputs a valid signature (r^*, s^*, σ^*) for message m^* and id^*, then σ^* is a valid signature for identity $\mathsf{CHash}(\mathsf{hk}, \mathsf{id}^*; r^*)$ and message $\mathsf{CHash}(\mathsf{hk}, \mathsf{m}^*; s^*)$ with respect to IBS'. Hence we only need to check freshness: For the sake of contradiction, assume the secret key of $\mathsf{CHash}(\mathsf{hk}, \mathsf{id}^*; r^*)$ was (non-adaptively) queried by $\mathcal{B}_2$. Then there is some $i \in [Q_C]$ such that $\mathsf{CHash}(\mathsf{hk}, \mathsf{id}^*; r^*) = \mathsf{CHash}(\mathsf{hk}, 0; \bar{r}_i)$. The way $\mathcal{B}_2$ answers signature queries tells us that $(\mathsf{CHash}(\mathsf{hk}, 0; \bar{r}_i), \mathsf{id}, r) \in \mathcal{H}_{id}$ for the i-th key query id and some randomness r. This is exactly the definition of event bad_1, which we ruled out before. An analogous argument using bad_2 shows that $(\mathsf{CHash}(\mathsf{hk}, \mathsf{id}^*; r^*), \mathsf{CHash}(\mathsf{hk}, \mathsf{m}^*; s^*))$ was not queried by $\mathcal{B}_2$. Thus, we have

$$\mathsf{Adv}_1(\mathcal{A}) \leq \mathsf{negl}(\lambda) + \mathsf{Adv}^{\mathsf{UF\text{-}naCMA}}_{\mathcal{B}_2, \mathsf{IBS}'}(\lambda).$$

Finally, the running time of $\mathcal{B}_2$ is dominated by evaluating the polynomial time chameleon hash and running adversary $\mathcal{A}$. □

3.2 Transformation in the Random Oracle Model

Next, we will show that if IBS' is non-adaptively secure and $\ell \in \omega(\log \lambda)$, then $\mathsf{IBS}_{\mathsf{ROM}}$, defined in Fig. 5, is adaptively secure. Clearly, if IBS' is ρ-complete, then $\mathsf{IBS}_{\mathsf{ROM}}$ is also ρ-complete.

Alg $\mathsf{KeyExt}(\mathsf{msk}, \mathsf{id})$
01 $r \xleftarrow{\$} \{0,1\}^\ell, \mathsf{id}' \leftarrow \mathsf{H}_1(\mathsf{id}, r)$
02 **return** $(r, \mathsf{KeyExt}'(\mathsf{msk}', \mathsf{id}'))$

Alg $\mathsf{Sig}(\mathsf{sk}_{\mathsf{id}} = (r, \mathsf{sk}'_{\mathsf{id}}), \mathsf{m})$
03 $s \xleftarrow{\$} \{0,1\}^\ell, \mathsf{m}' \leftarrow \mathsf{H}_2(\mathsf{m}, s)$
04 **return** $(r, s, \mathsf{Sig}'(\mathsf{sk}'_{\mathsf{id}}, \mathsf{m}'))$

Alg $\mathsf{Ver}(\mathsf{mpk}, \mathsf{id}, \mathsf{m}, \sigma)$
05 **let** $\sigma = (r, s, \sigma')$
06 $\mathsf{id}' \leftarrow \mathsf{H}_1(\mathsf{id}, r)$
07 $\mathsf{m}' \leftarrow \mathsf{H}_2(\mathsf{m}, s)$
08 $v := \mathsf{Ver}'(\mathsf{mpk}', \mathsf{id}', \mathsf{m}', \sigma')$
09 **return** v

Fig. 5. Our adaptively secure $\mathsf{IBS}_{\mathsf{ROM}} = (\mathsf{Setup} := \mathsf{Setup}', \mathsf{KeyExt}, \mathsf{Sig}, \mathsf{Ver})$ for a non-adaptively secure $\mathsf{IBS}' = (\mathsf{Setup}', \mathsf{KeyExt}', \mathsf{Sig}', \mathsf{Ver}')$ with random oracles $\mathsf{H}_1, \mathsf{H}_2$ and a natural number $\ell = \ell(\lambda)$.

Theorem 2. *Let* IBS' *be an identity-based signature scheme,* $\mathsf{H}_1 : \{0,1\}^* \to \mathcal{ID}, \mathsf{H}_2 : \{0,1\}^* \to \mathcal{M}$ *be random oracles and* $\ell = \ell(\lambda) \in \omega(\log(\lambda))$. *If* IBS' *is* UF-naCMA *secure, then* $\mathsf{IBS}_{\mathsf{ROM}}$ *is* UF-CMA *secure. In particular, for every algorithm* $\mathcal{A}$ *making at most* Q_S *signing queries,* Q_C *secret key queries and* Q_H *hash queries (including the indirect ones induced by singing and key queries) there is an algorithm* $\mathcal{B}$ *such that* $\mathbf{T}(\mathcal{B}) \approx \mathbf{T}(\mathcal{A})$ *and*

$$\mathsf{Adv}^{\mathsf{UF\text{-}CMA}}_{\mathcal{A},\mathsf{IBS}_{\mathsf{ROM}}}(\lambda) \leq \mathsf{Adv}^{\mathsf{UF\text{-}naCMA}}_{\mathcal{B},\mathsf{IBS}'}(\lambda) + (Q_C + 2Q_S)\frac{Q_H}{2^\ell} + \frac{Q_C}{|\mathcal{ID}|} + \frac{Q_S}{|\mathcal{ID}||\mathcal{M}|}.$$

Proof. We will sketch the proof via games $\mathbf{G}_0, \mathbf{G}_1, \mathbf{G}_2$ and denote the corresponding advantage of $\mathcal{A}$ in $\mathbf{G}_i$ by $\mathsf{Adv}_i(\mathcal{A}) := \Pr\left[\mathbf{G}_i^{\mathcal{A}} \Rightarrow 1\right]$. Game $\mathbf{G}_0$ is the original game **UF-CMA**.

Game $\mathbf{G}_1$ holds a variable bad_0 which is set to 1 whenever the hash value $\mathsf{H}_1(\mathsf{id}, r)$ or $\mathsf{H}_2(\mathsf{m}, s)$ in a signing query or a secret key query already has been defined. The game aborts if this variable is set. Thus it is clear that

$$|\mathsf{Adv}_0(\mathcal{A}) - \mathsf{Adv}_1(\mathcal{A})| \leq \Pr[\mathsf{bad}_0].$$

To analyze the probability of this event, we define events $\mathsf{bad}_{key,i}, i \in [Q_C]$ indicating that bad_0 was set to true in the i-th key query. Similarly we define the events $\mathsf{bad}_{sig,id,i}$ and $\mathsf{bad}_{sig,m,i}$ for $i \in [Q_S]$. The probabilities for these events can be bounded by $Q_H/2^\ell$ and we get

$$\begin{aligned}\Pr[\mathsf{bad}_0] &\leq \sum_{i\in[Q_C]} \Pr[\mathsf{bad}_{key,i}] + \sum_{i\in[Q_S]} (\Pr[\mathsf{bad}_{sig,id,i}] + \Pr[\mathsf{bad}_{sig,m,i}]) \\ &\leq (Q_C + 2Q_S)\frac{Q_H}{2^\ell}.\end{aligned}$$

In game $\mathbf{G}_2$ we add additional bad events $\mathsf{bad}_1, \mathsf{bad}_2$ (these are similar to the events in the proof of Thm. 1). If either of them occurs, the game will return 0. The game holds lists $\mathcal{H}_m$ and $\mathcal{H}_{id}$ such that $(\mathsf{id}', \mathsf{id}, r) \in \mathcal{H}_{id}$ means that the adversary asked for a secret key for identity id and when answering the query, the game hashed id to id' using r, i.e. $\mathsf{id}' = \mathsf{H}_1(\mathsf{id}; r)$. In a similar way, a tuple $((\mathsf{id}', \mathsf{id}, r), (\mathsf{m}', \mathsf{m}, s))$ is in $\mathcal{H}_m$ if in some signing query, id was hashed to id and m was hashed to m' using randomness r, s respectively. After obtaining $\mathcal{A}$'s forgery $(\mathsf{id}^*, \mathsf{m}^*, (r^*, s^*, \sigma^*))$, the game checks if the events occur:

$$\begin{aligned}\mathsf{bad}_1 &:= (\exists(\mathsf{id}', \mathsf{id}, r) \in \mathcal{H}_{id} : \mathsf{H}_1(\mathsf{id}^*, r^*) = \mathsf{id}') \\ \mathsf{bad}_2 &:= (\exists((\mathsf{id}', \mathsf{id}, r), (\mathsf{m}', \mathsf{m}, s)) \in \mathcal{H}_m : \mathsf{H}_1(\mathsf{id}^*, r^*) = \mathsf{id}' \wedge \mathsf{H}_2(\mathsf{m}^*, s^*) = \mathsf{m}').\end{aligned}$$

Similar to the corresponding step in the proof of Thm. 1, it is easy to see that the events imply collisions for H_1 or H_2. As $\mathcal{H}_{id}$ has Q_C entries, the probability of bad_1 can be upper bounded by $Q_C/|\mathcal{ID}|$. Similarly, the probability of bad_2 can be upper bounded by $Q_S(1/|\mathcal{M}| \cdot 1/|\mathcal{ID}|)$ and we obtain

$$|\mathsf{Adv}_1(\mathcal{A}) - \mathsf{Adv}_2(\mathcal{A})| \leq \Pr[\mathsf{bad}_1 \vee \mathsf{bad}_2] \leq \frac{Q_C}{|\mathcal{ID}|} + \frac{Q_S}{|\mathcal{ID}||\mathcal{M}|}.$$

Finally, we bound $\mathsf{Adv}_2(\mathcal{A})$ by a reduction $\mathcal{B}$ playing the game **UF-naCMA** for the scheme IBS'. Reduction $\mathcal{B}$ prepares all the variables needed to simulated $\mathbf{G}_2$ and then samples randomly

$$(\mathsf{id}'_i, \mathsf{m}'_i) \xleftarrow{\$} \mathcal{ID} \times \mathcal{M}, \mathsf{id}''_j \xleftarrow{\$} \mathcal{ID},$$

for all $i \in [Q_S], j \in [Q_C]$. Then it sends these to its challenger and obtains the public key mpk' as well as signatures σ'_i and secret keys sk'_j for these identities and messages. To answer the j-th key query id, $\mathcal{B}$ samples $r \xleftarrow{\$} \{0,1\}^\ell$, checks for the event bad as $\mathbf{G}_2$ does and programs the random oracle H_1 at (id, r) to be id''_j. It then returns sk'_j. This programming can be done without contradicting previous programming as we exactly ruled out that case by definition of bad. A similar programming strategy is applied for signing queries, programming both H_1 and H_2. In the end, after obtaining $\mathcal{A}$'s forgery $(\mathsf{id}^*, \mathsf{m}^*, (r^*, s^*, \sigma^*))$, reduction $\mathcal{B}$ simply applies random oracle to these values and outputs its own forgery

$$(\mathsf{H}_1(\mathsf{id}^*, r^*), \mathsf{H}_2(\mathsf{m}^*, s^*), \sigma^*).$$

Our definition of events $\mathsf{bad}_1, \mathsf{bad}_2$ makes sure that this forgery is fresh. Then it is clear that the running time of $\mathcal{B}$ is dominated by running $\mathcal{A}$ and we have

$$\mathsf{Adv}_2(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{UF\text{-}naCMA}}_{\mathcal{B},\mathsf{IBS}'}(\lambda).$$

□

4 Non-adaptive Security from SIS

In this section we construct a non-adaptively secure identity-based signature scheme $\mathsf{IBS}_{\mathsf{SIS}}$ based on the SIS assumption. Combined with the transformation presented in the previous section, we obtain an adaptively secure one. The scheme is presented in Fig. 6. Its main parameters are SIS parameters $n \in \mathbb{N}, q \in \mathbb{P}, m \geq 3n \log q, \beta > 0$. We also need Gaussian parameters $s_0, s, s', s'', \tilde{s} \geq \omega(\sqrt{\log m})$ (needed for regularity, Lem. 2), where the parameter $\tilde{s}$ is only used in the proof, s_0, s, s', s'' satisfy the conditions for Lem. 5 and β is large enough:

$$\begin{aligned}
s &\geq C_1 \sqrt{s_0^2 C_0^2 (\sqrt{m - n\lceil \log q \rceil} + \sqrt{n\lceil \log q \rceil})^2 + 1} \cdot \omega(\sqrt{\log n}) \\
s' &\geq C_1 \sqrt{s^2 C_0^2 (\sqrt{m} + \sqrt{n\lceil \log q \rceil})^2 + 1} \cdot \omega(\sqrt{\log n}) \\
s'' &\geq C_1 \sqrt{s'^2 C_0^2 (\sqrt{m + n\lceil \log q \rceil} + \sqrt{n\lceil \log q \rceil})^2 + 1} \cdot \omega(\sqrt{\log n}) \\
\beta &\geq (1 + 2C_0 \tilde{s}(\sqrt{m} + \sqrt{n\lceil \log q \rceil})) s'' \sqrt{m + 2n\lceil \log q \rceil}.
\end{aligned}$$

Lemma 6. *The identity-based signature scheme* $\mathsf{IBS}_{\mathsf{SIS}}$ *is* ρ*-complete, where* $\rho \geq 1 - \mathsf{negl}(\lambda)$.

Alg $\mathsf{Setup}(1^\lambda)$
01 set parameters as in the text.
02 $(\mathbf{A}, \mathbf{T}_\mathbf{A}) \leftarrow \mathsf{GenTrap}(1^n, 1^m, s_0, q)$
03 $\mathsf{mpk} := \mathbf{A} \in \mathbb{Z}_q^{n \times m}$, $\mathsf{msk} := \mathbf{T}_\mathbf{A}$
04 **return** $(\mathsf{mpk}, \mathsf{msk})$

Alg $\mathsf{Sig}(\mathsf{sk}_\mathsf{id}, \mathsf{m})$
05 $\mathbf{H}_1 \leftarrow \mathsf{H}_1(\mathsf{mpk}, \mathsf{id})$
06 $\mathbf{H}_2 \leftarrow \mathsf{H}_2(\mathsf{mpk}, \mathsf{id}, \mathsf{m})$
07 $\mathbf{F}_{\mathsf{id},\mathsf{m}} := [\mathbf{A} \mid \mathbf{H}_1 \mid \mathbf{H}_2]$
08 $\mathbf{T}_{\mathsf{id},\mathsf{m}} \leftarrow \mathsf{DelTrap}(\mathbf{F}_{\mathsf{id},\mathsf{m}}, \mathbf{T}_\mathsf{id}, s')$
09 $\mathbf{z} \leftarrow \mathsf{SampleD}(\mathbf{F}_{\mathsf{id},\mathsf{m}}, \mathbf{T}_{\mathsf{id},\mathsf{m}}, \mathbf{0}, s'')$
10 **return** $\sigma := \mathbf{z}$

Alg $\mathsf{KeyExt}(\mathsf{msk}, \mathsf{id})$
11 $\mathbf{H}_1 \leftarrow \mathsf{H}_1(\mathsf{mpk}, \mathsf{id})$
12 $\mathbf{F}_\mathsf{id} := [\mathbf{A} \mid \mathbf{H}_1]$
13 $\mathbf{T}_\mathsf{id} \leftarrow \mathsf{DelTrap}(\mathbf{F}_\mathsf{id}, \mathbf{T}_\mathbf{A}, s)$
14 **return** $\mathsf{sk}_\mathsf{id} := \mathbf{T}_\mathsf{id}$

Alg $\mathsf{Ver}(\mathsf{mpk}, \mathsf{id}, \mathsf{m}, \mathbf{z})$
15 $\mathbf{H}_1 \leftarrow \mathsf{H}_1(\mathsf{mpk}, \mathsf{id})$
16 $\mathbf{H}_2 \leftarrow \mathsf{H}_2(\mathsf{mpk}, \mathsf{id}, \mathsf{m})$
17 $\mathbf{F}_{\mathsf{id},\mathsf{m}} := [\mathbf{A} \mid \mathbf{H}_1 \mid \mathbf{H}_2]$
18 **if** $\mathbf{z} = \mathbf{0} \vee \mathbf{F}_{\mathsf{id},\mathsf{m}}\mathbf{z} \neq \mathbf{0}$: **return** 0
19 **return** $\|\mathbf{z}\| \leq s'' \sqrt{m + 2n\lceil \log q \rceil}$

Fig. 6. The identity-based signature scheme $\mathsf{IBS}_\mathsf{SIS} = (\mathsf{Setup}, \mathsf{KeyExt}, \mathsf{Sig}, \mathsf{Ver})$, where $\mathsf{H}_1 : \{0,1\}^* \to \mathbb{Z}_q^{n \times n\lceil \log q \rceil}, \mathsf{H}_2 : \{0,1\}^* \to \mathbb{Z}_q^{n \times n\lceil \log q \rceil}$ are random oracles.

Proof. Consider keys $(\mathsf{mpk} = \mathbf{A}, \mathsf{msk} = \mathbf{T}_\mathbf{A}) \in \mathsf{Setup}(1^\lambda)$, an arbitrary identity id and message m. Let $\mathsf{sk}_\mathsf{id} \in \mathsf{KeyExt}(\mathsf{msk}, \mathsf{id})$ and $\mathbf{z} \in \mathsf{Sig}(\mathsf{sk}_\mathsf{id}, \mathsf{m})$. By definition of KeyExt we have that $\mathsf{sk}_\mathsf{id} = \mathbf{T}_\mathsf{id}$ is a trapdoor for the matrix $\mathbf{F}_\mathsf{id} = [\mathbf{A} \mid \mathsf{H}_1(\mathsf{mpk}, \mathsf{id})]$, which is a prefix of $\mathbf{F}_{\mathsf{id},\mathsf{m}} = [\mathbf{A} \mid \mathsf{H}_1(\mathsf{mpk}, \mathsf{id}) \mid \mathsf{H}_2(\mathsf{mpk}, \mathsf{id}, \mathsf{m})]$. Hence $\mathbf{T}_{\mathsf{id},\mathsf{m}}$ as used in the signature scheme is a trapdoor for $\mathbf{F}_{\mathsf{id},\mathsf{m}}$, by Lem. 5 and the conditions of parameters. The same Lemma tells us that $\mathbf{z}$ is distributed statistically close to $D_{\Lambda_q^\perp(\mathbf{F}_{\mathsf{id},\mathsf{m}}), s''}$, which implies $\mathbf{F}_{\mathsf{id},\mathsf{m}}\mathbf{z} = \mathbf{0}$ and with overwhelming probability (by Lem. 3) $\|\mathbf{z}\| \leq s'' \cdot \sqrt{m + 2n\lceil \log q \rceil}$, which makes Ver accept. □

Theorem 3. *The scheme* $\mathsf{IBS}_\mathsf{SIS}$ *is an* $\mathsf{UF\text{-}naCMA}$ *secure identity-based signature scheme, under the* $\mathsf{SIS}_{n,m,q,\beta}$ *assumption. In particular, for every* PPT *algorithm* $\mathcal{A}$ *there is a* PPT *algorithm* $\mathcal{B}$ *such that*

$$\mathsf{Adv}^{\mathsf{UF\text{-}naCMA}}_{\mathcal{A},\mathsf{IBS}_\mathsf{SIS}}(\lambda) \leq \mathsf{Adv}^{\mathsf{SIS}_{n,m,q,\beta}}_{\mathcal{B}}(\lambda) + \mathsf{negl}(\lambda)$$

and $\mathbf{T}(\mathcal{B}) \approx \mathbf{T}(\mathcal{A})$.

Proof. The proof is via a reduction $\mathcal{B}$, formally given in Fig. 7. The idea is as follows: The given SIS matrix $\mathbf{A}$ is set as the master public key $\mathsf{mpk} := \mathbf{A}$. For any identity $\mathsf{id} \in \mathcal{L}_{id}$, for which the adversary queries the secret key, $\mathcal{B}$ programs the random oracle $\mathsf{H}_1(\mathsf{mpk}, \mathsf{id}) := \mathbf{A}\hat{\mathbf{R}}_{\mathsf{mpk},\mathsf{id}} + \mathbf{G}$, where $\hat{\mathbf{R}}_{\mathsf{mpk},\mathsf{id}} \leftarrow D_{\mathbb{Z},s}^{m \times n\lceil \log q \rceil}$ is short. Hence, $\hat{\mathbf{R}}_{\mathsf{mpk},\mathsf{id}}$ is a trapdoor for $\mathbf{F}_\mathsf{id} := [\mathbf{A} \mid \mathsf{H}_1(\mathsf{mpk}, \mathsf{id})]$ and $\mathcal{B}$ can return it as sk_id. Note that by definition of the non-adaptive security game, $\mathcal{A}$ did not query the random oracle before, hence programming is possible. For all other identities, the hash value will be programmed to $\mathsf{H}_1(\mathsf{mpk}, \mathsf{id}) := \mathbf{A}\hat{\mathbf{R}}_{\mathsf{mpk},\mathsf{id}}$. A similar programming policy is applied for H_2: For every pair $(\mathsf{id}, \mathsf{m}) \in \mathcal{L}_m$, for which the adversary wants to know a signature, the random oracle is programmed as $\mathsf{H}_2(\mathsf{mpk}, \mathsf{id}, \mathsf{m}) := \mathbf{A}\mathbf{R}_{\mathsf{mpk},\mathsf{id},\mathsf{m}} + \mathbf{G}$ for $\mathbf{R}_{\mathsf{mpk},\mathsf{id},\mathsf{m}} \leftarrow D_{\mathbb{Z},s}^{m \times n\lceil \log q \rceil}$. Using $\mathbf{R}_{\mathsf{mpk},\mathsf{id},\mathsf{m}}$ as a trapdoor for $[\mathbf{A} \mid \mathsf{H}_2(\mathsf{mpk}, \mathsf{id}, \mathsf{m})]$ the reduction can compute a trapdoor for $\mathbf{F}'_{\mathsf{id},\mathsf{m}} := [\mathbf{A} \mid \mathsf{H}_2(\mathsf{mpk}, \mathsf{id}, \mathsf{m}) \mid \mathsf{H}_1(\mathsf{mpk}, \mathsf{id})]$, sample from $D_{\Lambda_q^\perp(\mathbf{F}'_{\mathsf{id},\mathsf{m}}), s''}$ and

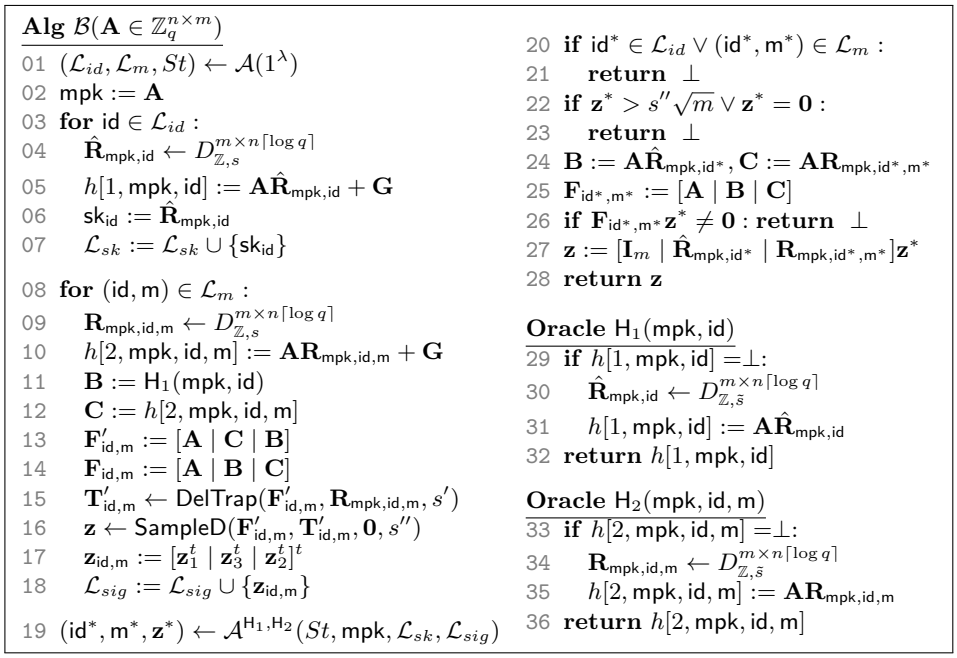

Fig. 7. Reduction $\mathcal{B}$, solving the SIS problem using an adversary $\mathcal{A}$ against the UF-naCMA security of $\mathsf{IBS}_{\mathsf{SIS}}$.

permute the resulting vector to get a signature as in the real scheme. Again, the random oracle value is not yet defined and can be programmed and other queries are programmed as $\mathsf{H}_2(\mathsf{mpk}, \mathsf{id}, \mathsf{m}) := \mathbf{A}\mathbf{R}_{\mathsf{mpk},\mathsf{id},\mathsf{m}}$. In the end $\mathcal{A}$ will return a forgery $(\mathsf{id}^*, \mathsf{m}^*, \mathbf{z}^*)$. We assume that $\mathcal{A}$ queried all related random oracle queries $\mathsf{H}_1(\mathsf{mpk}, \mathsf{id}^*)$ and $\mathsf{H}_2(\mathsf{mpk}, \mathsf{id}^*, \mathsf{m}^*)$ (otherwise we can build a new adversary making the queries after running $\mathcal{A}$ and having the same success probability). If $\mathcal{A}$ is successful, then $\mathsf{id}^* \notin \mathcal{L}_{id}$ and $(\mathsf{id}^*, \mathsf{m}^*) \notin \mathcal{L}_m$, which implies that

$$\mathbf{0} = \mathbf{F}_{\mathsf{id}^*,\mathsf{m}^*}\mathbf{z}^* = [\mathbf{A} \mid \mathbf{A}\hat{\mathbf{R}}_{\mathsf{mpk},\mathsf{id}^*} \mid \mathbf{A}\mathbf{R}_{\mathsf{mpk},\mathsf{id}^*,\mathsf{m}^*}]\mathbf{z}^*,$$

and hence $\mathcal{B}$ can return

$$\mathbf{z} := [\mathbf{I}_m \mid \hat{\mathbf{R}}_{\mathsf{mpk},\mathsf{id}^*} \mid \mathbf{R}_{\mathsf{mpk},\mathsf{id}^*,\mathsf{m}^*}]\mathbf{z}^*.$$

Note that the running time of the reduction is dominated by running $\mathcal{A}$. Let us now look at the details and show that the simulation is perfect (up to negligible statistical distance) and that $\mathbf{z}$ has the correct length and is not zero. As $s, \tilde{s} > \omega(\sqrt{\log m})$ and $m \geq 2n \log q$, Lem. 2 shows that $\mathbf{AR}$ is statistically close to uniform for $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$ and $\mathbf{R} \leftarrow D_{\mathbb{Z},\kappa}^{m \times n\lceil \log q \rceil}$ for $\kappa \in \{s, \tilde{s}\}$, hence the simulation of H_1 and H_2 is statistically close to the real game. Next, the distribution of the secret keys $\mathsf{sk}_{\mathsf{id}} = \hat{\mathbf{R}}_{\mathsf{mpk},\mathsf{id}}$ is statistically close to the real game, which can be obtained by applying the properties of DelTrap from Lem. 5.

Further, $\mathbf{R}_{\mathsf{mpk},\mathsf{id},\mathsf{m}}$ is a trapdoor for $[\mathbf{A} \mid \mathbf{A}\mathbf{R}_{\mathsf{mpk},\mathsf{id},\mathsf{m}} + \mathbf{G}]$, which is a prefix of $\mathbf{F}'_{\mathsf{id},\mathsf{m}}$ (as defined in Fig. 7) and thus the signature output by $\mathcal{B}$ is statistically close to $D_{\Lambda_q^\perp(\mathbf{F}_{\mathsf{id},\mathsf{m}}),s''}$, as honest signatures are. It remains to show that $\mathbf{z}$ is a suitable solution for SIS: To see that $\mathbf{z}$ is not zero, write $\mathbf{z}^* = [\mathbf{z}_1^* \mid \mathbf{z}_2^*]^t$ where $\mathbf{z}_1^* \in \mathbb{Z}_q^m, \mathbf{z}_2^* \in \mathbb{Z}_q^{2n\lceil \log q \rceil}$ and note that

$$\mathbf{z} = \mathbf{z}_1^* + [\hat{\mathbf{R}}_{\mathsf{mpk},\mathsf{id}^*} \mid \mathbf{R}_{\mathsf{mpk},\mathsf{id}^*,\mathsf{m}^*}]\mathbf{z}_2^*.$$

Now, if $\mathbf{z}_2^* = \mathbf{0}$, then $\mathbf{z} \neq \mathbf{0}$ as $\mathbf{z}_1^* \neq \mathbf{0}$. Otherwise there is some non-zero component $z_{2,j}^* \neq 0, j \in [2n\lceil \log q \rceil]$. Denote the columns of $[\hat{\mathbf{R}}_{\mathsf{mpk},\mathsf{id}^*} \mid \mathbf{R}_{\mathsf{mpk},\mathsf{id}^*,\mathsf{m}^*}]$ by $\mathbf{r}_i \in \mathbb{Z}_q^m, i \in [2n\lceil \log q \rceil]$. Then $\mathbf{z} = \mathbf{0}$ implies that

$$-\frac{1}{z_{2,j}^*}(\mathbf{z}_1^* + \sum_{i \neq j} z_{2,i}^* \mathbf{r}_i) = \mathbf{r}_j.$$

Further, note that the only information about $\mathbf{r}_j$ that $\mathcal{A}$ gets is a column of the hash value $\mathsf{H}_1(\mathsf{mpk}, \mathsf{id}^*) = \mathbf{A}\hat{\mathbf{R}}_{\mathsf{mpk},\mathsf{id}^*}$ (if $j \leq n\lceil \log q \rceil$) or $\mathsf{H}_2(\mathsf{mpk}, \mathsf{id}^*, \mathsf{m}^*) = \mathbf{A}\mathbf{R}_{\mathsf{mpk},\mathsf{id}^*,\mathsf{m}^*}$ (otherwise). Let $\mathbf{u}$ denote that column. Then from $\mathcal{A}$'s view, $\mathbf{r}_j$ is distributed as $D_{\Lambda_{\mathbf{u}}^\perp(\mathbf{A}),\tilde{s}}$. This distribution has a large min-entropy (with overwhelming probability over $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$) by Lem. 4, and hence the probability that $\mathbf{z} = \mathbf{0}$ is negligible. Finally, by Lem. 1 we have that $s_1(\hat{\mathbf{R}}_{\mathsf{mpk},\mathsf{id}^*}), s_1(\mathbf{R}_{\mathsf{mpk},\mathsf{id}^*,\mathsf{m}^*}) \leq C_0 \cdot \tilde{s} \cdot (\sqrt{m} + \sqrt{n\lceil \log q \rceil})$ with overwhelming probability. Hence

$$\begin{aligned}\|\mathbf{z}\| &\leq \|\mathbf{z}_1^*\| + \|[\hat{\mathbf{R}}_{\mathsf{mpk},\mathsf{id}^*} \mid \mathbf{R}_{\mathsf{mpk},\mathsf{id}^*,\mathsf{m}^*}]\mathbf{z}_2^*\| \\ &\leq (1 + 2C_0\tilde{s}(\sqrt{m} + \sqrt{n\lceil \log q \rceil}))s''\sqrt{m + 2n\lceil \log q \rceil} \leq \beta,\end{aligned}$$

which finishes the proof. □

Let us note the key and signature sizes (in bits) of our (non-adaptive) scheme :

$$\begin{aligned}|\mathsf{mpk}| &= n \cdot m \cdot \lceil \log(q) \rceil, & |\mathsf{msk}| &= (m - n\lceil \log(q) \rceil) \cdot n \cdot \lceil \log(q) \rceil^2, \\ |\mathsf{sk}_{\mathsf{id}}| &= m \cdot n \cdot \lceil \log(q) \rceil^2, & |\sigma| &= (m + 2 \cdot n \cdot \lceil \log(q) \rceil) \cdot \lceil \log(q) \rceil.\end{aligned}$$

In order to get concrete parameters, we can use an estimation that in every delegation, it is enough (up to constants) that the Gaussian width multiplies with $\sqrt{m} \cdot \omega(\sqrt{\log m})$. For the worst-case to average-case reductions [42,22] to work, we need to choose $q \geq \beta \cdot \mathsf{poly}(n)$, where $\mathsf{poly}(n)$ can grow roughly as $\sqrt{n}$. One obtains that the following example instantiation satisfies all our conditions for large enough n and $n^4 \leq q \leq n^5$ prime:

$$\begin{aligned}m &:= 3n \log q, & s_0 &:= \tilde{s} := \omega(\sqrt{\log m}), \\ s &:= \hat{C} \cdot m^{1/2} \cdot \omega(\sqrt{\log m})^2, & s' &:= \hat{C}^2 \cdot m \cdot \omega(\sqrt{\log m})^3, \\ s'' &:= \hat{C}^3 \cdot m^{3/2} \cdot \omega(\sqrt{\log m})^4, & \beta &:= \tilde{C} \cdot n^{5/2} \cdot \log(n)^{5/2} \cdot \omega(\sqrt{\log m})^5,\end{aligned}$$

where $\hat{C} := 4C_0C_1$ and $\tilde{C} := 48 \cdot 3^{3/2} \cdot 5^{5/2} \cdot \hat{C}^3 C_0$ are constants chosen such that the estimation is correct.

References

1. Abdalla, M., Fouque, P.A., Lyubashevsky, V., Tibouchi, M.: Tightly-secure signatures from lossy identification schemes. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 572–590. Springer, Heidelberg (Apr 2012) 2
2. Ajtai, M.: Generating hard instances of lattice problems (extended abstract). In: 28th ACM STOC. pp. 99–108. ACM Press (May 1996) 1, 3, 8, 9
3. Bader, C., Hofheinz, D., Jager, T., Kiltz, E., Li, Y.: Tightly-secure authenticated key exchange. In: Dodis, Y., Nielsen, J.B. (eds.) TCC 2015, Part I. LNCS, vol. 9014, pp. 629–658. Springer, Heidelberg (Mar 2015) 2
4. Banerjee, A., Peikert, C., Rosen, A.: Pseudorandom functions and lattices. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 719–737. Springer, Heidelberg (Apr 2012) 3
5. Bellare, M., Boldyreva, A., Micali, S.: Public-key encryption in a multi-user setting: Security proofs and improvements. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 259–274. Springer, Heidelberg (May 2000) 1
6. Bellare, M., Namprempre, C., Neven, G.: Security proofs for identity-based identification and signature schemes. In: Cachin, C., Camenisch, J. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 268–286. Springer, Heidelberg (May 2004) 2
7. Bellare, M., Ristenpart, T.: Simulation without the artificial abort: Simplified proof and improved concrete security for Waters' IBE scheme. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 407–424. Springer, Heidelberg (Apr 2009) 1
8. Blazy, O., Kakvi, S.A., Kiltz, E., Pan, J.: Tightly-secure signatures from chameleon hash functions. In: Katz, J. (ed.) PKC 2015. LNCS, vol. 9020, pp. 256–279. Springer, Heidelberg (Mar / Apr 2015) 2, 3
9. Blazy, O., Kiltz, E., Pan, J.: (Hierarchical) identity-based encryption from affine message authentication. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part I. LNCS, vol. 8616, pp. 408–425. Springer, Heidelberg (Aug 2014) 2
10. Boyen, X., Li, Q.: Towards tightly secure lattice short signature and id-based encryption. In: Cheon, J.H., Takagi, T. (eds.) ASIACRYPT 2016, Part II. LNCS, vol. 10032, pp. 404–434. Springer, Heidelberg (Dec 2016) 2, 3, 4
11. Cash, D., Hofheinz, D., Kiltz, E., Peikert, C.: Bonsai trees, or how to delegate a lattice basis. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 523–552. Springer, Heidelberg (May / Jun 2010) 3, 4, 5, 7
12. Chen, J., Wee, H.: Fully, (almost) tightly secure IBE and dual system groups. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 435–460. Springer, Heidelberg (Aug 2013) 1, 2
13. Cramer, R., Damgård, I., Schoenmakers, B.: Proofs of partial knowledge and simplified design of witness hiding protocols. In: Desmedt, Y. (ed.) CRYPTO'94. LNCS, vol. 839, pp. 174–187. Springer, Heidelberg (Aug 1994) 2
14. del Pino, R., Lyubashevsky, V., Neven, G., Seiler, G.: Practical quantum-safe voting from lattices. In: Thuraisingham, B.M., Evans, D., Malkin, T., Xu, D. (eds.) ACM CCS 2017. pp. 1565–1581. ACM Press (Oct / Nov 2017) 2
15. Diemert, D., Gellert, K., Jager, T., Lyu, L.: More efficient digital signatures with tight multi-user security. In: PKC 2021 (2021), https://ia.cr/2021/235 2
16. Dodis, Y., Katz, J., Xu, S., Yung, M.: Strong key-insulated signature schemes. In: Desmedt, Y. (ed.) PKC 2003. LNCS, vol. 2567, pp. 130–144. Springer, Heidelberg (Jan 2003) 2

17. Ducas, L., Lepoint, T., Lyubashevsky, V., Schwabe, P., Seiler, G., Stehle, D.: CRYSTALS – Dilithium: Digital signatures from module lattices. Cryptology ePrint Archive, Report 2017/633 (2017), http://eprint.iacr.org/2017/633 5
18. Fouque, P.A., Hoffstein, J., Kirchner, P., Lyubashevsky, V., Pornin, T., Prest, T., Ricosset, T., Seiler, G., Whyte, W., Zhang, Z.: Falcon: Fast-fourier lattice-based compact signatures over ntru. Submission to the NIST's post-quantum cryptography standardization process 36 (2018) 5
19. Gay, R., Hofheinz, D., Kohl, L.: Kurosawa-desmedt meets tight security. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017, Part III. LNCS, vol. 10403, pp. 133–160. Springer, Heidelberg (Aug 2017) 2
20. Gay, R., Hofheinz, D., Kohl, L., Pan, J.: More efficient (almost) tightly secure structure-preserving signatures. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part II. LNCS, vol. 10821, pp. 230–258. Springer, Heidelberg (Apr / May 2018) 1, 2
21. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. Cryptology ePrint Archive, Report 2007/432 (2007), http://eprint.iacr.org/2007/432 8
22. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: Ladner, R.E., Dwork, C. (eds.) 40th ACM STOC. pp. 197–206. ACM Press (May 2008) 4, 5, 8, 9, 18
23. Gentry, C., Silverberg, A.: Hierarchical ID-based cryptography. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 548–566. Springer, Heidelberg (Dec 2002) 2
24. Gjøsteen, K., Jager, T.: Practical and tightly-secure digital signatures and authenticated key exchange. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018, Part II. LNCS, vol. 10992, pp. 95–125. Springer, Heidelberg (Aug 2018) 2
25. Goh, E.J., Jarecki, S., Katz, J., Wang, N.: Efficient signature schemes with tight reductions to the Diffie-Hellman problems. Journal of Cryptology 20(4), 493–514 (Oct 2007) 3
26. Goldwasser, S., Micali, S., Rivest, R.L.: A digital signature scheme secure against adaptive chosen-message attacks. SIAM Journal on Computing 17(2), 281–308 (Apr 1988) 2
27. Groth, J., Sahai, A.: Efficient non-interactive proof systems for bilinear groups. In: Smart, N.P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 415–432. Springer, Heidelberg (Apr 2008) 2
28. Hofheinz, D.: Algebraic partitioning: Fully compact and (almost) tightly secure cryptography. In: Kushilevitz, E., Malkin, T. (eds.) TCC 2016-A, Part I. LNCS, vol. 9562, pp. 251–281. Springer, Heidelberg (Jan 2016) 2
29. Hofheinz, D., Jager, T.: Tightly secure signatures and public-key encryption. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 590–607. Springer, Heidelberg (Aug 2012) 2
30. Jager, T., Kurek, R., Pan, J.: Simple and more efficient PRFs with tight security from LWE and matrix-DDH. In: Peyrin, T., Galbraith, S. (eds.) ASIACRYPT 2018, Part III. LNCS, vol. 11274, pp. 490–518. Springer, Heidelberg (Dec 2018) 3
31. Katsumata, S., Yamada, S., Yamakawa, T.: Tighter security proofs for GPV-IBE in the quantum random oracle model. In: Peyrin, T., Galbraith, S. (eds.) ASIACRYPT 2018, Part II. LNCS, vol. 11273, pp. 253–282. Springer, Heidelberg (Dec 2018) 2, 4
32. Kiltz, E., Neven, G.: Identity-based signatures. In: Joye, M., Neven, G. (eds.) Identity-Based Cryptography. IOS Press (2009) 2, 5

This page is exceeding the page limit.

33. Kim, S.: Key-homomorphic pseudorandom functions from LWE with small modulus. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part II. LNCS, vol. 12106, pp. 576–607. Springer, Heidelberg (May 2020) 3
34. Krawczyk, H., Rabin, T.: Chameleon signatures. In: NDSS 2000. The Internet Society (Feb 2000) 4
35. Langrehr, R., Pan, J.: Tightly secure hierarchical identity-based encryption. In: Lin, D., Sako, K. (eds.) PKC 2019, Part I. LNCS, vol. 11442, pp. 436–465. Springer, Heidelberg (Apr 2019) 2
36. Langrehr, R., Pan, J.: Hierarchical identity-based encryption with tight multi-challenge security. In: Kiayias, A., Kohlweiss, M., Wallden, P., Zikas, V. (eds.) PKC 2020, Part I. LNCS, vol. 12110, pp. 153–183. Springer, Heidelberg (May 2020) 2
37. Langrehr, R., Pan, J.: Unbounded HIBE with tight security. In: Moriai, S., Wang, H. (eds.) ASIACRYPT 2020, Part II. LNCS, vol. 12492, pp. 129–159. Springer, Heidelberg (Dec 2020) 2
38. Lee, Y., Park, J.H., Lee, K., Lee, D.H.: Tight security for the generic construction of identity-based signature (in the multi-instance setting). Theoretical Computer Science 847, 122–133 (2020), https://www.sciencedirect.com/science/article/pii/S0304397520305557 2
39. Lyubashevsky, V.: Lattice signatures without trapdoors. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 738–755. Springer, Heidelberg (Apr 2012) 5
40. Micciancio, D., Peikert, C.: Trapdoors for lattices: Simpler, tighter, faster, smaller. Cryptology ePrint Archive, Report 2011/501 (2011), http://eprint.iacr.org/2011/501 8
41. Micciancio, D., Peikert, C.: Trapdoors for lattices: Simpler, tighter, faster, smaller. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 700–718. Springer, Heidelberg (Apr 2012) 3, 4, 8, 9
42. Micciancio, D., Regev, O.: Worst-case to average-case reductions based on Gaussian measures. In: 45th FOCS. pp. 372–381. IEEE Computer Society Press (Oct 2004) 8, 9, 18
43. Pan, J., Wagner, B.: Short identity-based signatures with tight security from lattices. In: Cheon, J.H., Tillich, J.P. (eds.) Post-Quantum Cryptography. pp. 360–379. Springer International Publishing, Cham (2021) 1
44. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Gabow, H.N., Fagin, R. (eds.) 37th ACM STOC. pp. 84–93. ACM Press (May 2005) 8
45. Shamir, A.: Identity-based cryptosystems and signature schemes. In: Blakley, G.R., Chaum, D. (eds.) CRYPTO'84. LNCS, vol. 196, pp. 47–53. Springer, Heidelberg (Aug 1984) 2
46. Zhang, X., Liu, S., Gu, D., Liu, J.K.: A generic construction of tightly secure signatures in the multi-user setting. Theoretical Computer Science 775, 32–52 (2019), https://www.sciencedirect.com/science/article/pii/S0304397518307333 5