Lattice-based Blind Signatures

Markus Rückert*
rueckert@cdc.informatik.tu-darmstadt.de

Technische Universität Darmstadt
Department of Computer Science
Cryptography and Computeralgebra

Abstract. Motivated by the need to have secure blind signatures even in the presence of quantum computers, we present two efficient blind signature schemes based on hard worst-case lattice problems. Both schemes are provably secure in the random oracle model and unconditionally blind. The first scheme is based on preimage samplable functions that were introduced at STOC 2008 by Gentry, Peikert, and Vaikuntanathan. The scheme is stateful and runs in 3 moves. The second scheme builds upon the PKC 2008 identification scheme of Lyubashevsky. It is stateless, has 4 moves, and its security is based on the hardness of worst-case problems in ideal lattices.

Keywords. Blind signatures, post-quantum, lattices

*This work was supported by CASED (www.cased.de).

1 Introduction

Since 1982, when Chaum proposed his idea of blind signatures [Cha83], it has become an important primitive for anonymous Internet banking, e-voting applications (e.g. [RHOAGZ07, Kim04]), as well as for multi-party computation such as oblivious transfer [CNS07]. These applications will retain their importance in both, near and far future. As for the near future, we are convinced that current factoring and discrete logarithm based instantiations are efficient and secure. *But for how long?*

Today, when building provably secure cryptographic schemes, one has to keep emerging technologies and especially quantum computers in mind. In the quantum-age, the cryptographic assumptions change with the leap in computing power that quantum computers will provide.

There are only a few cryptographic assumptions that are conjectured to be *post-quantum*, i.e. they are considered to withstand quantum computer attacks. One of those assumptions is the hardness of finding short vectors in a lattice. There is also a benefit of building cryptography upon hard lattice problems today because, unlike factoring or computing discrete logarithms, they have withstood even subexponential attacks and the best known algorithm [AKS01] is exponential in the lattice dimension. Furthermore, lattice problems typically allow a worst-case to average-case reduction that goes back to Ajtai [Ajt96]. It states that a randomly chosen instance of a certain lattice problem is at least as hard as the worst-case instance of a related lattice problem. The reduction was later on adapted to work with ideal lattices by Lyubashevsky and Micciancio [LM06].

According to the security model, mainly influenced by Juels, Luby, and Ostrovsky [JLO97] as well as Pointcheval and Stern [PS00], blind signature schemes have to satisfy blindness and one-more unforgeability. Blindness states that the signer must not obtain any information on the signed messages and one-more unforgeability enforces that an adversarial user cannot obtain more signatures than there were interactions with the signer.

1.1 Our contribution

We construct two lattice-based blind signatures. One is based on preimage samplable functions that were introduced by Gentry, Peikert, and Vaikuntanathan (GPV) [GPV08] along with a digital signature scheme. The scheme is stateful, unconditionally blind, one-more unforgeable if a certain interactive assumption (similar to the one-more trapdoor inversion assumption in [BNPS03] for RSA) holds, and has three moves. The scheme is presented using general (not ideal) lattices but recently Stehlé, Seinfeld, Tanaka, and Xagawa [SSTX09] showed how to improve the GPV signature scheme using ideal lattices. Their modifications are directly applicable to our blind signature scheme and significantly reduce the public-key size.

Our second construction is far stronger. It is built upon Lyubashevsky's identification and signature scheme [Lyu08b, Lyu08a]. It is also unconditionally blind and one-more unforgeable if standard lattice problems in ideal lattices are hard in the worst-case. With its four rounds it is still very efficient, i.e., all operations have quasi-linear complexity and all keys and signatures require a quasi-linear amount of bits. In both schemes, we establish blindness via an abstraction of the filtering technique from [Lyu08a].

We believe that our work is an important contribution and that we solve a longstanding problem because the previous efficient constructions, like [Cha83], [PS97], [PS00], [Abe01], [BNPS03], [CKW04], [KZ05], [Oka06], have one thing in common: they are built upon classic number theoretic assumptions, like the hardness of factoring large integers or computing discrete logarithms. The

newer approaches of Boldyreva [Bol03] and Okamoto [Oka06] tend to use pairings and bilinear maps that yield very elegant constructions. They, however, are again based on the discrete logarithm problem in this specific setting. None of the above schemes withstands subexponential attacks or remains secure in the presence of reasonably large quantum computers, where both factoring and computing discrete logarithms become easy due to the seminal work of Shor [Sho97].

Finally, we would like to mention that there are also (typically inefficient) instantiations from general assumptions, e.g. by Juels, Luby, and Ostrovsky [JLO97], Fischlin [Fis06], and Hazay, Katz, Koo, and Lindell [HKKL07]. Whether they are post-quantum, largely depends on the exact realization of primitives.

1.2 Organization

After a preliminaries section with a brief introduction to lattice theory and the relevant security models, we present our constructions in Sections 3 and 4. The instantiation in Section 3 is based on a trapdoor function in lattices, whereas the one in Section 4 is based on an identification scheme in ideal lattices. There, we also prove that our scheme has the well-established security properties.

2 Preliminaries

With n, we always denote the security parameter. $(a, b) \leftarrow \langle \mathcal{A}(x), \mathcal{B}(y) \rangle$ denotes the joint execution of two algorithms $\mathcal{A}$ and $\mathcal{B}$ in an interactive protocol with private inputs x to $\mathcal{A}$ and y to $\mathcal{B}$. The private outputs are a for $\mathcal{A}$ and b for $\mathcal{B}$. $\langle \mathcal{A}(x), \mathcal{B}(y) \rangle^k$ means that the interaction can take place up to k times.

$x \xleftarrow{\$} X$ means that x is chosen uniformly at random from the finite set X. Recall that the statistical distance of two random variables X, Y over a domain D is defined as $\Delta(X, Y) = 1/2 \sum_{a \in D} |\operatorname{Prob}[X = a] - \operatorname{Prob}[Y = a]|$. A function is negligible in n if it vanishes faster than $1/p(n)$ for any polynomial $p(n)$.

In the following, we recall the definitions of blind signatures and commitments. Afterwards, we briefly recall the forking lemma and some necessary facts from lattice theory.

2.1 Blind signatures

A blind signature scheme BS consists of three algorithms $(\mathsf{Kg}, \mathsf{Sign}, \mathsf{Vf})$, where Sign is an interactive protocol between a signer $\mathcal{S}$ and a user $\mathcal{U}$. The specification is as follows.

Key generation. $\mathsf{Kg}(1^n)$ outputs a private signing key sk and a public verification key pk.

Signature issse. $\mathsf{Sign}(\mathsf{sk}, M)$ describes the joint execution of $\mathcal{S}$ and $\mathcal{U}$. The private output of $\mathcal{S}$ is a view $\mathcal{V}$ and the private output of $\mathcal{U}$ is a signature $\mathbf{s}$ on the message $M \in \mathcal{M}$ under sk. Thus, we write $(\mathcal{V}, \mathbf{s}) \leftarrow \langle \mathcal{S}(\mathsf{sk}), \mathcal{U}(\mathsf{pk}, M) \rangle$.

Signature verification. The algorithm $\mathsf{Vf}(\mathsf{pk}, \mathbf{s}, M)$ outputs 1 if $\mathbf{s}$ is a valid signature on M under pk and otherwise 0.

Completeness is defined as with digital signature schemes, i.e., every honestly created signature for honestly created keys and for any messages $M \in \mathcal{M}$ has to be valid under this key. Views are interpreted as random variables, whose output is generated by subsequent executions of the

respective protocol. Two views $\mathcal{V}_1$ and $\mathcal{V}_2$ are considered equal if they cannot be distinguished by any computationally unbounded algorithm with noticeable probability.

As for security, blind signatures have to satisfy two properties: blindness and one-more unforgeability [JLO97, PS00]. The notion of blindness is defined in the following experiment $\mathsf{Exp}^{\mathsf{blind}}_{\mathcal{S}^*,\mathsf{BS}}$, where the adversarial signer $\mathcal{S}^*$ chooses two messages M_0, M_1 and interacts with two users who obtain blind signatures for the two messages in random order. Note that the executions of the two users may be arbitrarily interleaved. After seeing the unblinded signatures in the original order, with respect to M_0, M_1, the signer has to guess the message that has been signed for the first user. If either of the user algorithms fails in outputting a valid signature, the signer is merely notified of the failure and does not get any signature. In particular, he does not see which user algorithm aborted.

Experiment $\mathsf{Exp}^{\mathsf{blind}}_{\mathcal{S}^*,\mathsf{BS}}(n)$

$b \xleftarrow{\$} \{0,1\}$
$(\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{BS.Kg}(1^n)$
$(M_0, M_1, \mathsf{state}_{\mathrm{find}}) \leftarrow \mathcal{S}^*(\mathrm{find}, \mathsf{sk}, \mathsf{pk})$
$(d, \mathsf{state}_{\mathrm{issue}}) \leftarrow \mathcal{S}^{*\langle\cdot,\mathcal{U}(\mathsf{pk},M_b)\rangle^1,\langle\cdot,\mathcal{U}(\mathsf{pk},M_{1-b})\rangle^1}(\mathrm{issue}, \mathsf{state}_{\mathrm{find}})$
Let $\mathbf{s}_b$ and $\mathbf{s}_{1-b}$ be the outputs of $\mathcal{U}(\mathsf{pk}, M_b)$ and $\mathcal{U}(\mathsf{pk}, M_{1-b})$, respectively.
If $\mathbf{s}_0 \neq \mathsf{fail}$ and $\mathbf{s}_1 \neq \mathsf{fail}$
  $d \leftarrow \mathcal{S}^*(\mathrm{guess}, \mathbf{s}_0, \mathbf{s}_1, \mathsf{state}_{\mathrm{issue}})$
Else
  $d \leftarrow \mathcal{S}^*(\mathrm{guess}, \mathsf{fail}, \mathsf{fail}, \mathsf{state}_{\mathrm{issue}})$
Return 1 iff $d = b$

A signature scheme BS is (t,δ)-blind, if there is no adversary $\mathcal{S}^*$, running in time at most t, that wins the above experiment with advantage at least δ, where the advantage is defined as

$$\mathsf{Adv}^{\mathsf{blind}}_{\mathcal{S}^*,\mathsf{BS}} = \left|\mathrm{Prob}\left[\mathsf{Exp}^{\mathsf{blind}}_{\mathcal{S}^*,\mathsf{BS}}(n) = 1\right] - \frac{1}{2}\right| .$$

The second security property, one-more unforgeability, ensures that each completed interaction between signer and user yields at most one signature. It is formalized in the following experiment $\mathsf{Exp}^{\mathsf{omf}}_{\mathcal{U}^*,\mathsf{BS}}$, where an adversarial user tries to output $\jmath$ valid signatures after $\ell < \jmath$ completed interactions with an honest signer.

Experiment $\mathsf{Exp}^{\mathsf{omf}}_{\mathcal{U}^*,\mathsf{BS}}(n)$

$\mathsf{H} \xleftarrow{\$} \mathcal{H}(1^n)$
$(\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{BS.Kg}(1^n)$
$\{(M_1,\mathbf{s}_1),\ldots,(M_\jmath,\mathbf{s}_\jmath)\} \leftarrow \mathcal{U}^{*\mathsf{H}(\cdot),\langle\mathcal{S}(\mathsf{sk}),\cdot\rangle}(\mathsf{pk})$
Let ℓ be the number of (complete) interaction between $\mathcal{U}^*$ and the signer.
Return 1 iff
  1. $M_i \neq M_j$ for all $1 \leq i < j \leq \jmath$;
  2. $\mathsf{BS.Vf}(\mathsf{pk}, \mathbf{s}_i, M_i) = 1$ for all $i = 1,\ldots,\jmath$;
  3. $\ell < \jmath$.

A signature scheme BS is $(t, q_{\mathsf{Sign}}, q_{\mathsf{H}}, \delta)$-one-more unforgeable if there is no adversary $\mathcal{A}$, running in time at most t, making at most q_{Sign} signature queries and at most q_{H} hash oracle queries, that wins the above experiment with probability at least δ.

2.2 Commitments

Commitments typically work in two phases. First, one party publishes a commitment $C = \mathsf{com}(M; r)$ to a message M without revealing any information about it. This is the "hiding" property of the commitment scheme. In the second phase, the party can prove that C actually corresponds to M by revealing r. It is important that no algorithm can find a second message M' and randomness r' such that $C = \mathsf{com}(M'; r')$ — the "binding" property. A scheme is (t, δ)-hiding (-binding) if there is no algorithm running in time at most t that can break the hiding (binding) property with probability at least δ.

Both properties can be satisfied computationally or unconditionally but there is no scheme that is unconditionally hinding *and* unconditionally binding [Gol04]. For our schemes, we want blindness to be as strong as possible, which is why we assume the existence of a unconditionally hiding and computationally binding commitment scheme that is $(t, \delta_{\mathsf{com}})$-binding for any polynomial t in n.

As we are interested in fully lattice-based schemes, we would like to point out that commitment schemes can be built upon on hard lattice problems [KTX08].

2.3 Forking Lemma

The generalized forking lemma of Bellare and Neven [BN06] is an important tool for proving security in the random oracle model. It provides a lower bound for the probability that a randomized algorithm outputs two related values when run twice with the same random tape but with a different random oracle. We use it in Section 4 to prove one-more unforgeability.

Lemma 2.1 (Lemma 1 in [BN06]) *Fix an integer $q \geq 1$ and a set H of size $h \geq 2$. Let A be a randomized algorithm that on input $x, h_1, \ldots, h_q$ returns a pair, the first element of which is an integer in the range $0, \ldots, q$ and the second element of which we refer to as a side output. Let IG be a randomized algorithm that we call the input generator. The accepting probability of A, denoted* acc*, is defined as the probability that $J \geq 1$ in the experiment*

$$x \xleftarrow{\$} \mathsf{IG}; h_1, \ldots, h_q \xleftarrow{\$} H; (J, \sigma) \xleftarrow{\$} \mathsf{A}(x, h_1, \ldots, h_q)\,.$$

The forking algorithm F_{A} associated to A is the randomized algorithm that takes input x proceeds as follows:

Algorithm $\mathsf{F}_{\mathsf{A}}(x)$
Pick coins ρ for A at random
$h_1, \ldots, h_q \xleftarrow{\$} H$
$(I, \sigma) \leftarrow \mathsf{A}(x, h_1, \ldots, h_q; \rho)$
If $I = 0$ then return $(0, \epsilon, \epsilon)$
$h'_I, \ldots, h'_q \xleftarrow{\$} H$
$(I', \sigma') \leftarrow \mathsf{A}(x, h_1, \ldots, h_{I-1}, h'_I, \ldots, h'_q; \rho)$
If $I = I'$ and $h_I \neq h'_{I'}$ then return $(1, \sigma, \sigma')$
Else return $(0, \epsilon, \epsilon)$.

Let

$$\mathsf{frk} = \mathrm{Prob}\left[b = 1 : x \xleftarrow{\$} \mathsf{IG}; (b, \sigma, \sigma')\mathsf{F}_{\mathsf{A}}(x)\right]\,.$$

Then

$$\mathsf{frk} \geq acc\left(\frac{acc}{q} - \frac{1}{h}\right).$$

2.4 Lattices

A lattice in $\mathbb{R}^n$ is a set $\Lambda = \{\sum_{i=1}^{d} x_i \mathbf{b}_i \,|\, x_i \in \mathbb{Z}\}$, where $\mathbf{b}_1, \ldots, \mathbf{b}_d$ are linearly independent over $\mathbb{R}$. The matrix $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_d]$ is a *basis* of the lattice Λ and we write $\Lambda = \Lambda(\mathbf{B})$. The number of linearly independent vectors in the basis is the dimension of the lattice. Now, consider *modular lattices* as a special form of lattices. Given a modulus q, a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, and the equation $\mathbf{A}\,\mathbf{v} \equiv \mathbf{0} \pmod{q}$, then the set of all vectors $\mathbf{v} \in \mathbb{Z}_q^m$ that satisfy the above equation is a lattice. Lattices of this form are denoted with $\Lambda_q^{\perp}(\mathbf{A})$.

The main computational problem in lattices is the (approximate) shortest vector problem (SVP), where an algorithm is given a description, a basis, of a lattice Λ and is supposed to find the shortest vector $\mathbf{v} \in \Lambda \backslash \{\mathbf{0}\}$ with respect to a certain ℓ_p norm (up to an approximation factor). More precisely, find a vector $\mathbf{v} \in \Lambda \setminus \{\mathbf{0}\}$, such that $\|\mathbf{v}\|_p \leq \gamma \|\mathbf{w}\|_p$ for all $\mathbf{w} \in \Lambda \setminus \{\mathbf{0}\}$ for a fixed approximation factor $\gamma \geq 1$. This problem is known to be $\mathcal{NP}$-hard for all ℓ_p norms [Din02, RR06, Kho05] with a constant approximation factor. For exponential (in the lattice dimension) approximation factors, the problem is solvable in polynomial time by the famous LLL algorithm by Lenstra, Lenstra, and Lovász [LLL82]. For polynomial approximation factors, which are relevant for cryptography, the best known algorithm is exponential (space and time) [AKS01]. We refer the interested reader to a recent survey [Reg07] by Regev for the currently known "approximability" and "inapproximability" results. The practical hardness of these lattice problems is analyzed in [GN08, BLR08].

In the special case of modular lattices, there is also a special version of the SVP, named short integer solution problem (SIS). There, an algorithm is given a basis of $\Lambda_q^{\perp}(\mathbf{A})$ and is supposed to output a non-zero solution $\mathbf{v} \in \mathbb{Z}_q^m$ to the above equation. The algorithm succeeds if $\|\mathbf{v}\|_p \leq \nu$ for a given norm bound ν. The SIS was, in principle, introduced by Ajtai [Ajt96] and its hardness is analyzed in [MR07] and [GPV08]. The latter work also explicitly deals with the ℓ_∞ norm, which we will use in our security proofs. We write $\mathsf{SIS}^p(m, q, \nu)$ for the SIS problem in m-dimensional lattices $\Lambda_q^{\perp}(\mathbf{A})$ with norm bound ν w.r.t. the ℓ_p norm. The problem is (t, δ)-hard if no algorithm that runs in time t can solve it with probability at least δ. Similarly, the inhomogeneous version (find a short $\mathbf{v}$ with $\mathbf{A}\mathbf{v} \equiv \mathbf{y}$, for a given $\mathbf{y} \neq \mathbf{0}$) is denoted with $\mathsf{ISIS}^p(m, q, \nu)$.

Yet another special class of lattices are ideal lattices. In particular, consider lattices corresponding to ideals in the ring $\mathbf{R} = \mathbb{Z}_q[X]/\langle X^n + 1\rangle$. We identify $\mathbf{f} \in \mathbf{R}$ with its coefficient vector $\mathbf{f} = (f_0, \ldots, f_{n-1}) \in \mathbb{Z}_q^n$. Furthermore, we denote elements of the $\mathbf{R}$-module $\mathbf{R}^m$ with $\hat{\mathbf{a}} = (\mathbf{a_0}, \ldots, \mathbf{a_{m-1}})$ or directly with $(a_0, \ldots, a_{mn-1}) \in \mathbb{Z}_q^{mn}$. Consequently, we define $\|\mathbf{f}\|_\infty = \|(f_0, \ldots, f_{n-1})\|_\infty$. A lattice corresponds to an ideal I if and only if every lattice vector is the coefficient vector of a polynomial in I. The above problems SIS and ISIS easily translate to ideal lattices.

Both, ideal SIS and SIS are considered as average-case problems, which are directly related to uniformly random chosen problem instances in lattice cryptography. By a worst-case to average-case reduction [Ajt96, LM06] they are provably at least as hard as *all* instances of ideal SVP (ISVP) resp. SVP in a certain smaller dimension.

3 Blind signatures from preimage samplable functions

In this section, we describe our blind signature scheme and prove its security in terms of *blindness* and *one-more unforgeability*. It is based on the signature scheme by Gentry et al. [GPV08] and we describe it in terms of general lattices. However, following the ideas of [SSTX09], there is also a version using ideal lattices, which has a significantly shorter public keys.

The roadmap for this section is as follows: We describe the 3-round blind signature scheme $\mathsf{BS} = (\mathsf{Kg}, \mathsf{Sign}, \mathsf{Vf})$ after briefly recalling the concept of preimage samplable functions as they will be needed in our construction. Then, we prove unconditional blindness and one-more unforgeability based on an interactive assumption that is related to a certain lattice prolem but *not* equivalent.

The underlying family of *preimage samplable trapdoor functions* is a triple $(\mathsf{TrapGen}, \mathsf{SampleDom}, \mathsf{SamplePre})$, with the following specification.

Trapdoor generation. $\mathsf{TrapGen}(1^n)$ outputs (a, t), where a fully defines the function $f_a : D_n \mapsto R_n$ and the trapdoor t is used to sample from the inverse $f_t^{-1} : R_n \mapsto D_n^\star$, which is implemented as $\mathsf{SamplePre}(t, \cdot)$. Let $m = 5n\log(q)$, $q = \Omega(n^3)$, and $D = \omega(m\log(m))$. The function domain is $D_n = \{\mathbf{x} \in \mathbb{Z}^m : \|\mathbf{x}\|_\infty \leq xmD - D\}$, $x \in \mathbb{N}_{>0}$, and the range is $R_n = \mathbb{Z}_q^n$. $\mathsf{SamplePre}$ samples preimages from a subset $D_n^\star = \{\mathbf{x} \in \mathbb{Z}^m : \|\mathbf{x}\|_\infty \leq D\}$ of D_n.

Evaluation. The function $f_a(\mathbf{x})$ outputs $\mathbf{A}\mathbf{x} \mod q$, where $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ is part of the public key a.

Domain sampling with uniform output. $\mathsf{SampleDom}(n)$ draws samples from some distribution over $D_n^\star$, such that their images under f_a are uniformly distributed over R_n(cf. [GPV08]).

Preimage sampling. Let $\mathbf{y} \in R_n$. $f_t^{-1}(\mathbf{y})$ samples $\mathbf{x} \leftarrow \mathsf{SampleDom}(n)$ under the condition that $f_a(\mathbf{x}) = \mathbf{y}$. There are at least $\omega(\log(n))$ such preimages for every image $\mathbf{y}$.

One-wayness. Computing an inverse of the function $f_a : D_n \mapsto R_n$ is infeasible without the trapdoor t as long as $\mathsf{ISIS}^\infty(m, q, xmD - D)$ is hard.

Collision resistance. Finding a collision $(\mathbf{x}, \mathbf{x}') \in D_n^2$ under f_a is infeasible unless $\mathsf{SIS}^\infty(m, q, 2xmD - 2D)$ is easy.

Note that we slightly modified the original setting regarding the sets $D_n, D_n^\star$. In [GPV08], it is always the same, whereas we have introduced different $D_n, D_n^\star$ for trapdoor evaluation and preimage sampling, respectively. As in the original work, we will always assume that the above properties, especially the statistical distributions, hold for f_a in a perfect sense. Using a proposition from [GPV08], we can establish the following corollary for our choice of parameters:

Corollary 3.1 *Let n, m, q, D, x as above. If there is a polynomial time (in n) algorithm that breaks* SIS *with ν (or* ISIS *with ν) with non-negligible probability then there is another polynomial time algorithm that solves* SIVP *(a variant of* SVP*) with approximation factors $\gamma \geq \nu\widetilde{\mathcal{O}}(\sqrt{n})$ in all lattices of dimension n.*

In addition to the above trapdoor function, we need a full-domain hash function (cf. [BR93]) $\mathsf{H} \leftarrow \mathcal{H}(1^n)$, where $\mathsf{H} : \{0,1\}^* \to R_n$ and $\mathcal{H}$ is a family of collision resistant hash functions. We assume that there is no polynomial time algorithm that finds collisions but with negligible probability δ_H.

Our blind signature scheme $\mathsf{BS} = (\mathsf{Kg}, \mathsf{Sign}, \mathsf{Vf})$ is defined as follows.

Key generation. $\mathsf{BS.Kg}(1^n)$ outputs $(a,t) \leftarrow \mathsf{TrapGen}(1^n)$, where a is the public verification key and t is the secret signing key, and sets up a list of already signed messages $L_M = \{\mathbf{0}\} \subseteq R_n$.

Signature protocol. Let $D_\beta = \{\mathbf{x} \in \mathbb{Z}^m : \|\mathbf{x}\|_\infty \leq xmD\}$. The signature issue protocol for messages $M \in \{0,1\}^*$ is shown in Figure 1. The user employs a commitment scheme $\mathsf{com} : \{0,1\}^* \times \{0,1\}^n \to \{0,1\}^*$ that is unconditionally hiding and computationally binding (but with probability δ_{com}). Note that the blind signature scheme is stateful, i.e. the signer does not sign a blinded message μ twice and it does not sign $\mu = \mathbf{0} \in R_n$ in particular[1]. The result is $\mathbf{s} \in D_n$.

Verification. $\mathsf{BS.Vf}(a,(r,\mathbf{s}),M)$ outputs 1 iff $\mathbf{s} \in D_n$ and $f_a(\mathbf{s}) = \mathsf{H}(\mathsf{com}(M;r))$.

Completeness. The scheme BS is complete with constant probability $e^{-1/x}$ because for all honestly generated key pairs (a,t), all messages $M \in \{0,1\}^*$, and all signatures $(r,\mathbf{s})$, we have $\mathbf{s} = \sigma - \beta$ and $f_a(\mathbf{s}) = f_a(\sigma - \beta) = f_a(\sigma) - f_a(\beta) = f_a(f_t^{-1}(\mathsf{H}(\mathsf{com}(M;r)) + f_a(\beta))) - f_a(\beta) = \mathsf{H}(\mathsf{com}(M;r))$. Assuming $\sigma \in D_n$, we also have $\mathsf{BS.Vf}(a,\mathbf{s},M) = 1$. This happens with constant probability as shown in the following probabilistic lemma with $k = n, A = D, B = xmD$.

Lemma 3.2 *Let $k \in \mathbb{N}$ and $\mathbf{a}, \mathbf{b} \in \mathbb{Z}^k$ with*

$$\begin{aligned} \mathbf{a} &\in \{\mathbf{v} \in \mathbb{Z}^k : \|\mathbf{v}\|_\infty \leq A\} \\ \mathbf{b} &\xleftarrow{\$} \{\mathbf{v} \in \mathbb{Z}^k : \|\mathbf{v}\|_\infty \leq B\} \end{aligned}$$

and $B \geq xkA$ for $x \in \mathbb{N}_{>0}$. Then

$$\Prob_{\mathbf{b}}[\,\|\mathbf{a}-\mathbf{b}\|_\infty \leq B - A] > \frac{1}{e^{1/x}} - o(1).$$

Proof. Observe that $\Prob[\,\|\mathbf{a}-\mathbf{b}\|_\infty \leq B-A] = \Prob[\,|a_i - b_i| \leq B - A]^k$ and that b_i needs to be in the range $[-(B-A)+a_i, B-A+a_i] \subseteq [-B,B]$. Therefore, the probability for this event is

$$\left(\frac{2(B-A)+1}{2B+1}\right)^k > \left(1 - \frac{A}{B}\right)^k \geq \left(1 - \frac{1/x}{k}\right)^k > \frac{1}{e^{1/x}} - o(1).$$

□

Setting $x \geq 2$, we expect the protocol to be complete in a single run. If the protocol fails, the user simply reveals the current interaction $(\mathsf{com}(M;r), \beta, \sigma)$ to the signer in order to prove that the execution failed. If the commitment scheme is perfectly hiding, the user does not reveal any information about M. Then, the protocol is repeated with fresh values for r and β. Observe that this does not affect the upcoming security analysis because the individual protocol runs are indepedent. In particular, the hiding property of com can be directly used in the blindness proof and the binding property is used in the proof of unforgeability.

[1] Signing $\mathbf{0}$ would result in a short vector in $\Lambda_q^\perp(A)$ and help learn the private signing key similar to the method in [NR06]. Due to the linearity of f_a, the same applies if a message is signed twice.

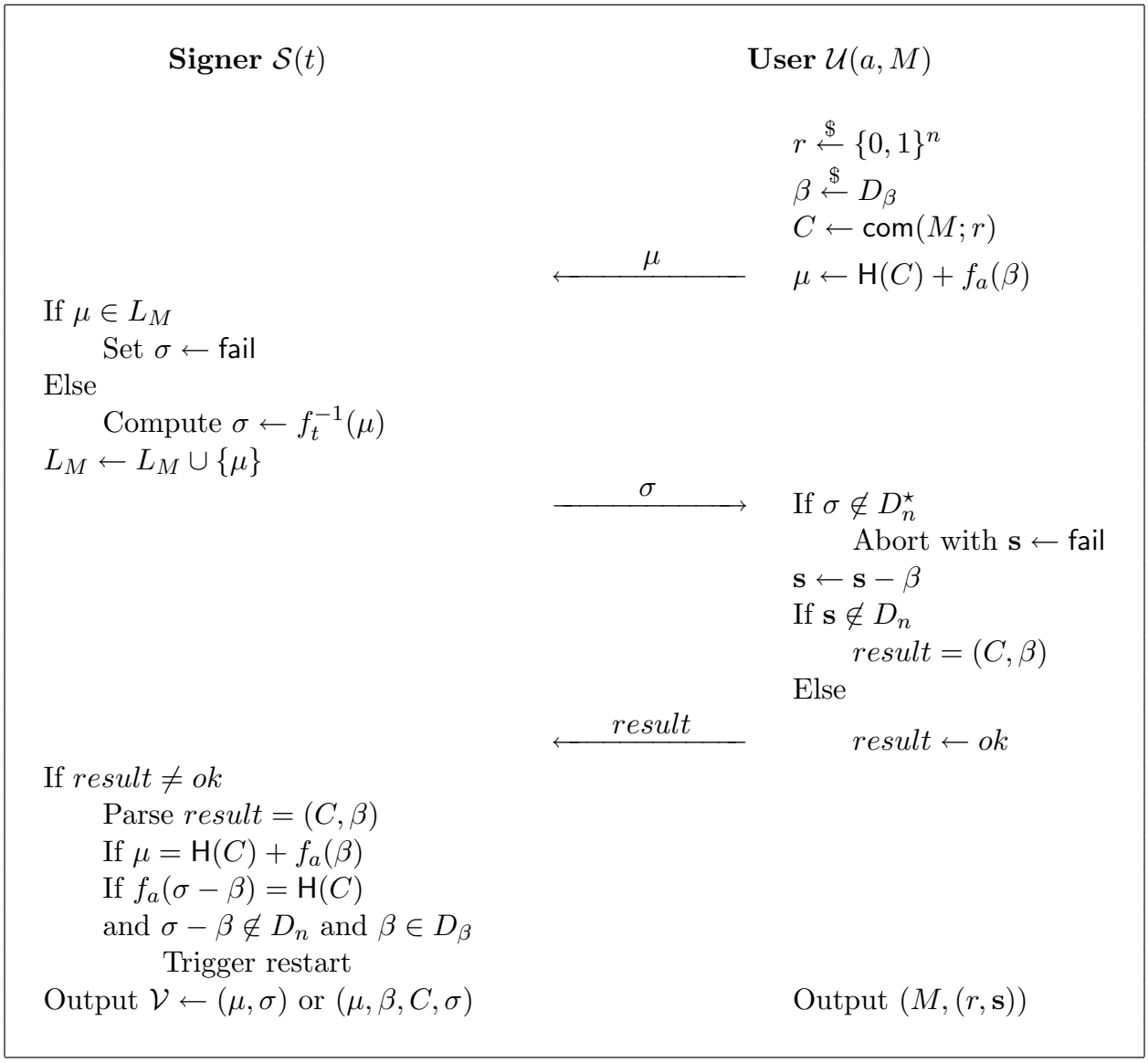

Figure 1: Issue protocol of the blind signature scheme BS.

Blindness. We prove that BS is unconditionally blind, i.e. $(\infty, 0)$-blind, if com is unconditionally hiding. If it is only statistically or computationally hiding, the blind signature scheme is also statistically resp. computationally hiding. The intuition is that the signer only sees random elements from R_n after the user has applied a random blinding value. The output signature is again randomized by a sufficiently large value β, which hides the internal ordinary signature.

Theorem 3.3 (Blindness) *The blind signature scheme* BS *is* $(\infty, 0)$*-blind.*

Proof.

The idea of the proof is that, given the signer's view in the experiment $\mathsf{Exp}^{\mathsf{blind}}_{\mathcal{S}^*,\mathsf{BS}}$, it cannot relate M and μ and it cannot distinguish the whether σ hides a signature on M_0 or M_1. The former is due to the fact that f_a is regular [Lyu08a] for uniformly random chosen inputs $\beta \in D_\beta$ but with $2^{-\omega(n)}$ statistical distance from uniform over R_n. The latter is a consequence of the following probabilistic lemma (Lemma 3.4) with $k = n, A = D, B = xmD$. Furthermore, since the commitment scheme is unconditionally hiding, the signer cannot learn anything about the message by aborting the protocol because the views for independent runs for the same message are completely independent as the user does not ever reveal the randomness r for a failed interaction and the blinded message μ is always uniformly distributed over R_n. Thus, we infer that the signer can only guess b with probability 1/2.

Lemma 3.4 *Let $k \in \mathbb{N}$ and $\mathbf{a}, \mathbf{a}', \mathbf{b} \in \mathbb{Z}^k$ with*

$$\begin{aligned} \mathbf{a}, \mathbf{a}' &\in \{\mathbf{v} \in \mathbb{Z}^k : \|\mathbf{v}\|_\infty \leq A\} \\ \mathbf{b} &\xleftarrow{\$} \{\mathbf{v} \in \mathbb{Z}^k : \|\mathbf{v}\|_\infty \leq B\} \end{aligned}$$

such that $\|a - b\|_\infty, \|a' - b\|_\infty \leq B - A$ and $B > A$. Then

$$\Delta(\mathbf{a} - \mathbf{b}, \mathbf{a}' - \mathbf{b}) = 0.$$

Proof (Lemma 3.4). By definition, the statistical distance is

$$\begin{aligned} \Delta(\mathbf{a} - \mathbf{b}, \mathbf{a}' - \mathbf{b}) &= \sum_{\mathbf{c} \in \mathbb{Z}^k : \|\mathbf{c}\|_\infty \leq B - A} \left| \Prob_{\mathbf{b}}[\mathbf{a} - \mathbf{b} = \mathbf{c}] - \Prob_{\mathbf{b}}[\mathbf{a}' - \mathbf{b} = \mathbf{c}] \right| \\ &= \sum_{\mathbf{c}} \left| \Prob_{\mathbf{b}}[\mathbf{b} = \mathbf{a} - \mathbf{c}] - \Prob_{\mathbf{b}}[\mathbf{b} = \mathbf{a}' - \mathbf{c}] \right|. \end{aligned}$$

Observe that $\|\mathbf{a} - \mathbf{c}\|_\infty, \|\mathbf{a}' - \mathbf{c}\|_\infty \leq A + (B - A) = B$ and $\|\mathbf{b}\|_\infty \leq B$. Hence, the probability in either case is $1/(2B + 1)^k$ and the statistical distance is 0. □

In the context of blindness, the above lemma states that the signer cannot distinguish whether a given blind signature σ is of the form $\mathbf{s}_b - \beta$ or $\mathbf{s}_{1-b} - \beta$, where $\mathbf{s}_b$ is valid for M_b and $\mathbf{s}_{1-b}$ is valid for M_{1-b}. This conludes the proof of blindness. □

One-more unforgeability. We prove that our blind signature scheme is unforgeable under a special assumption, namely that the following "one-more trapdoor inversion problem" is hard.

Definition 3.5 (Chosen target trapdoor inversion problem (CTTI)) *The chosen target trapdoor inversion problem is defined via the following experiment $\mathsf{Exp}^{\mathsf{ctti}}_{\mathcal{A}}$, where the adversary $\mathcal{A}$ has access to a challenge oracle O_{R_n} and to an inversion oracle f_t^{-1}. The adversary wins, if it outputs $\jmath$ preimages for challenges obtained from O_{R_n}, while making only $\imath < \jmath$ queries to f_t^{-1}. The oracle f_t^{-1} does not answer queries twice and its does not invert $\mathbf{0} \in R_n$ and it returns preimages in $D_n^\star$.*

Experiment $\mathsf{Exp}^{\mathsf{ctti}}_{\mathcal{A}}(n)$
$(a, t) \leftarrow \mathsf{TrapGen}(1^n)$
$(\pi, \mathbf{x}_1, \ldots, \mathbf{x}_\jmath) \leftarrow \mathcal{A}^{\mathsf{O}_{R_n}, f_t^{-1}(\cdot)}(n, a)$

Note: f_t^{-1} does not answer to $\mathbf{0}$ or already queried values.
Let $\mathbf{y}_1, \ldots, \mathbf{y}_\ell$ be the challenges returned by O_{R_n}.
Let $\imath$ be the number of queries to f_t^{-1}.
Return 1 iff

1. The $\mathbf{x}_i$ are pairwise distinct and
2. $\|\mathbf{x}_i\|_\infty \leq xmD + D$ and $f_a(\mathbf{x}_i) = \mathbf{y}_{\pi(i)}$ for all $i = 1, \ldots, \jmath$ and
3. $\imath < \jmath$.

The problem is $(t, q_\mathsf{I}, q_\mathsf{O}, \delta)$-hard if there is no algorithm $\mathcal{A}$, running in time at most t, making at most q_I inversion queries, and at most q_O queries to O_{R_n}, which wins the above experiment with probability at least δ. The one-wayness of f_a gives us $(\mathsf{poly}(n), 0, 1, \delta)$-hardness, which we will extend to $(\mathsf{poly}(n), \mathsf{poly}(n), \mathsf{poly}(n), \delta')$-hardness for a negligible δ'. With our definition and this assumption, we follow the line of thought of Bellare, Namprempre, Pointcheval, and Semanko in [BNPS03]. They define a collection of "one-more" problems in the RSA context, which are perfectly tailored for proving one-more unforgeability. In [BMV08], Bresson, Monnerat, and Vergnaud give a separation result on these "one-more" problems, showing that they cannot be proven equivalent to "simple" RSA inversion. The same seems to apply here. There is also a recent work on so-called *adaptive one-way functions* by Pandey, Pass, and Vaikuntanathan [PPV08], which discusses similar assumptions.

In the following, we will assume $(\mathsf{poly}(n), \mathsf{poly}(n), \mathsf{poly}(n), \delta)$-hardness of CTTI on the grounds that it is directly related to the provably hard problem of forging GPV signatures. In both cases, one has to find a solution $\mathbf{x} \in D_n$ to the equation $f_a(\mathbf{x}) = \mathbf{y}$ for a given $\mathbf{y}$, while knowing polynomially many distinct preimage-image pairs.

Theorem 3.6 (One-more unforgeability) *Let T_Sig and T_H be the cost functions for simulating the oracles* Sig *and* H*, respectively. The* BS *blind signature scheme is $(t, q_\mathsf{Sign}, q_\mathsf{H}, \delta)$-one-more unforgeable if the* CTTI *is $(t, q_\mathsf{Sig}, q_\mathsf{H}, \delta - \delta_\mathsf{H} - \delta_\mathsf{com})$-hard.*

Proof. Towards contradiction, we assume that there exists a successful forger $\mathcal{A}$ against one-more unforgeability of BS. Using $\mathcal{A}$, we construct an algorithm $\mathcal{B}$ via a black-box simulation, such that $\mathcal{B}$ solves the respective instance of the CTTI. The simulation works as follows.

Setup. $\mathcal{B}$ gets as input the public trapdoor parameter a and has access to the challenge oracle O_{R_n} and to a trapdoor inversion oracle f_t^{-1}. $\mathcal{B}$ initializes a list $L_\mathsf{H} \leftarrow \emptyset$ of pairs $(M, \mathbf{c})$, indexed by M, a list $L_\mathsf{I} \leftarrow \emptyset$ of pairs (μ, σ), indexed by μ, and two counters $\ell \leftarrow 0$, $\imath \leftarrow 0$. It runs $\mathcal{A}$ on input a in a black-box simulation.

Random oracle queries. On input M, $\mathcal{B}$ looks up M in L_H. If it finds a pair $(M, \mathbf{c})$ then it returns c. Otherwise, $\mathcal{B}$ increments $\imath$, chooses a new $\mathbf{c}_\imath \leftarrow \mathsf{O}_{R_n}$, stores $(M_\imath \leftarrow M, \mathbf{c}_\imath)$ in L_H. Afterwards, $\mathcal{B}$ returns $\mathbf{c}_\imath$.

Blind signature queries. On input μ, algorithm $\mathcal{B}$ searches a pair (μ, σ) in L_I. If it exists, $\mathcal{B}$ returns σ. Otherwise, algorithm $\mathcal{B}$ increments ℓ, queries its inversion oracle $\sigma_\ell \leftarrow f_t^{-1}(\mu)$, stores $(\mu_\ell \leftarrow \mu, \sigma_\ell)$ in L_I, and returns σ_ℓ.

Output. Finally, $\mathcal{A}$ stops and outputs $((M_1, (r_1, \mathbf{s}_1)), \ldots, (M_\jmath, (r_\jmath, \mathbf{s}_\jmath)))$, $\ell < \jmath$, for distinct messages. W.l.o.g., assume that $(\mathsf{com}(M_i; r_i), c_i) \in L_\mathsf{H}$, for all $i = 1, \ldots, \jmath$. Algorithm $\mathcal{B}$ sets $\pi' = \{(i, j) :$

$f_a(\mathbf{s}_i) = c_j\}$. Furthermore, it collects all views $(\mu'_{j+1}, \beta'_{j+1}, C'_{j+1}, \sigma'_{j+1}), \ldots, (\mu'_k, \beta'_k, C'_k, \sigma'_k)$ for which the protocol was aborted and sets $\pi'' = \{(i,j) : f_a(\sigma'_i - \beta_i) = \mathsf{H}(C'_j)\}$ to account for the inversions that are not used by $\mathcal{A}$. It outputs $(\pi' \cup \pi'', \mathbf{s}_1, \ldots, \mathbf{s}_j, \sigma_{j+1} - \beta_{j+1}, \ldots, \sigma_k - \beta_k)$.

Analysis. First, observe that all of $\mathcal{A}$'s oracles are perfectly simulated. When $\mathcal{A}$ calls H, algorithm $\mathcal{B}$ draws a new challenge from its challenge oracle. Whenever $\mathcal{A}$ queries its signature oracle on a new blinded message, $\mathcal{B}$ calls its inversion oracle. Therefore, when $\mathcal{A}$ outputs a one-more forgery, $\mathcal{B}$ can use it to solve CTTI.

Firstly, we have to rule out that the user has an additional advantage via user-side aborts. Consider the case, where the user outputs a (fake) $result = (C', \beta')$ that satisfies the abort conditions $\mathsf{H}(C') + f_a(\beta') = \mu$ and $\sigma - \beta' \notin D_n$, while obtaining a valid signature $\mathbf{s} = \sigma - \beta \in D_n$ for $C = \mathsf{com}(M; r)$. If $C = C'$, the simulation outputs two distinct preimages $\sigma - \beta$ and $\sigma - \beta'$ for C, which is valid in the CTTI experiment. If $C \neq C'$, the simulations records $\sigma - \beta'$ as a preimage for $\mathsf{H}(C')$ and outputs it at the end and $\mathsf{H}(C) \neq \mathsf{H}(C')$ but with probability δ_{H}. Hence, even with aborts, $\mathcal{B}$ is able to output the required number of preimages for solving CTTI.

Secondly, assume that all the $\mathbf{s}_1, \ldots, \mathbf{s}_j$ are distinct. Then, $\mathcal{B}$'s output is valid in the CTTI experiment because all preimages are small and evaluate to challenges received from O_{R_n} and the number of output inversions k is greater than the number of inversion queries ℓ and from with the previous paragraph we know that aborts contribute valid additional preimages that are different from the $\mathbf{s}_1, \ldots, \mathbf{s}_j$.

Finally, assume there is a pair $(M, (r, \mathbf{s})), (M', (r', \mathbf{s}))$ in $\mathcal{A}$'s output with $M \neq M'$ such that $\mathsf{H}(\mathsf{com}(M; r)) = f_a(\mathbf{s}) = \mathsf{H}(\mathsf{com}(M'; r'))$. That either implies a collision under H or a successful attack against the binding property of com.

Thus, if $\mathcal{A}$ succeeds with probability δ, $\mathcal{B}$ succeeds with probabiltiy $\delta - \delta_{\mathsf{H}} - \delta_{\mathsf{com}}$. □

4 Blind signatures from ideal lattices

In this section, we construct a second lattice-based blind signature scheme. Here, the construction is not built upon a trapdoor, which allows us to fully simulate the scheme in our security proofs and give very strong arguments for one-more unforgeability. The underlying signature scheme is due to Lyubashevsky [Lyu08b]. Both, Lyubashevsky's signature scheme and our blind signature scheme are secure in the random oracle model under a worst-case assumption in ideal lattices and their time *and* space complexity is only $\widetilde{\mathcal{O}}(n)$.

The roadmap for this section is as follows: We describe the 4-round blind signature scheme $\mathsf{BS} = (\mathsf{Kg}, \mathsf{Sign}, \mathsf{Vf})$. Then, we prove unconditional blindness and one-more unforgeability based on the assumptions that solving ISVP in dimension n is hard in the worst case.

For the setup, we need the global parameters in Table 1, where $\mathbf{R} = \mathbb{Z}_p[X]/\langle X^n + 1\rangle$. The scheme relies on the lattice-based collision resistant hash function family $\mathcal{H}(\mathbf{R}, D, m)$ by Lyubashevsky and Micciancio [LM06]. We fix a random $h \xleftarrow{\$} \mathcal{H}(\mathbf{R}, D, m)$, mapping $D^m \mapsto \mathbf{R}$, $D \subset \mathbf{R}$. Note that the function is linear over $\mathbf{R}^m$, i.e., $h(\mathbf{a}(\hat{\mathbf{x}} + \hat{\mathbf{y}}) = \mathbf{a}(h(\hat{\mathbf{x}}) + h(\hat{\mathbf{y}}))$ for all $\mathbf{a} \in \mathbf{R}$, $\hat{\mathbf{x}}, \hat{\mathbf{y}} \in \mathbf{R}^m$. In addition, finding a collision $(x, x') \in D^2$ under h, i.e. solving $Col(h, D)$ or alternatively ideal SIS^∞ with $\nu = |D|^{1/n} - 1$, implies being able to solve ISVP^∞ in every lattice that corresponds to an ideal in $\mathbf{R}$. More formally, from [Lyu08b], we know:

Parameter	Value
m	$\log(n)$
p (prime)	$\geq 4n^2m\log^2(n)(x^3n^3m^2 - x^2n^2m^2 - 2x^2n^2m + 2xnm + xn) = \Theta(n^5\log^5(n))$
D_s, D_ϵ	$\{\mathbf{f} \in \mathbf{R} : \|\mathbf{f}\|_\infty \leq 1\}$
D_α	$\{\mathbf{f} \in \mathbf{R} : \|\mathbf{f}\|_\infty \leq xn\}$ for a constant $x \in \mathbb{N}_{>0}$
D_{ϵ^*}	$\{\mathbf{f} \in \mathbf{R} : \|\mathbf{f}\|_\infty \leq xn - 1\}$
D_y	$\{\mathbf{f} \in \mathbf{R} : \|\mathbf{f}\|_\infty \leq \sqrt{n}\log(n)(x^2n^2m - xnm)\}$
D_β	$\{\mathbf{f} \in \mathbf{R} : \|\mathbf{f}\|_\infty \leq \sqrt{n}\log(n)(x^3n^3m^2 - x^2n^2m^2 - x^2n^2m + xnm)\}$
G_*	$\{\mathbf{f} \in \mathbf{R} : \|\mathbf{f}\|_\infty \leq \sqrt{n}\log(n)(x^2n^2m - xnm - xn + 1)\}$
G	$\{\mathbf{f} \in \mathbf{R} : \|\mathbf{f}\|_\infty \leq \sqrt{n}\log(n)(x^3n^3m^2 - x^2n^2m^2 - 2x^2n^2m + 2xnm + xn - 1)\}$
D	$\{\mathbf{f} \in \mathbf{R} : \|\mathbf{f}\|_\infty \leq \sqrt{n}\log(n)(x^3n^3m^2 - x^2n^2m^2 - xn + 1)\}$

Table 1: Parameters for the security parameter n.

Theorem 4.1 (Theorem 3.1 in [Lyu08b]) *Let $D = \{\mathbf{f} \in \mathbf{R} : \|\mathbf{f}\|_\infty \leq d\}$, $m > \log(p)/\log(2d)$, and $p \geq 4dmn\sqrt{n}\log(n)$. An adversary $\mathcal{C}$ that solves the $Col(h, D)$ problem, i.e., finds two preimages $\hat{\mathbf{x}}, \hat{\mathbf{y}} \in D^m$ such that $h(\hat{\mathbf{x}}) = h(\hat{\mathbf{y}})$, can be used to solve ISVP^∞ with an approximation factor of $\gamma \geq 16dmn\log^2(n)$ in* every *lattice that corresponds to an ideal in $\mathbb{Z}[X]/\langle\mathbf{f}\rangle$.*

Furthermore, we need a random oracle $\mathsf{H} \xleftarrow{\$} \mathcal{H}(1^n)$ mapping $\{0,1\}^* \mapsto D_\epsilon$. Again, there is no polynomial time algorithm that finds collisions but with negligible probability δ_H.

Key generation. $\mathsf{BS.Kg}(1^n)$ selects a secret key $\hat{\mathbf{s}} \xleftarrow{\$} D_s^m$ and computes the public key $\mathbf{S} \leftarrow h(\hat{\mathbf{s}})$. The output is $(\hat{\mathbf{s}}, \mathbf{S})$.

Signature protocol. The signature issue protocol for messages $M \in \{0,1\}^*$ is depicted in Figure 2. Note that values controlled by the user are written as Greek letter and those controlled by the signer are in Latin. In the first step, the user employs a commitment scheme $\mathsf{com} : \{0,1\}^* \times \{0,1\}^n \to \{0,1\}^*$ that we assume to be unconditionally hiding and computationally binding (but with probability δ_com).

Whenever the signer triggers a restart, the user chooses a fresh r in order to make the protocol execution independent of the previous one. Therefore, we omit values from previous runs in the signer's view. The signer can also detect a cheating user that tries to trigger a restart although it has received a valid signature. In this case, the signer can stop the protocol and assume that the user has obtained a valid signature.

Eventually, the user outputs $(r, \hat{\mathbf{z}}, \epsilon)$.

Verification. $\mathsf{BS.Vf}(a, (r, \hat{\mathbf{z}}, \epsilon), M)$ outputs 1 iff $\hat{\mathbf{z}} \in G^m$ and $\mathsf{H}(h(\hat{\mathbf{z}}) - \mathbf{S}\epsilon, \mathsf{com}(M; r)) = \epsilon$.

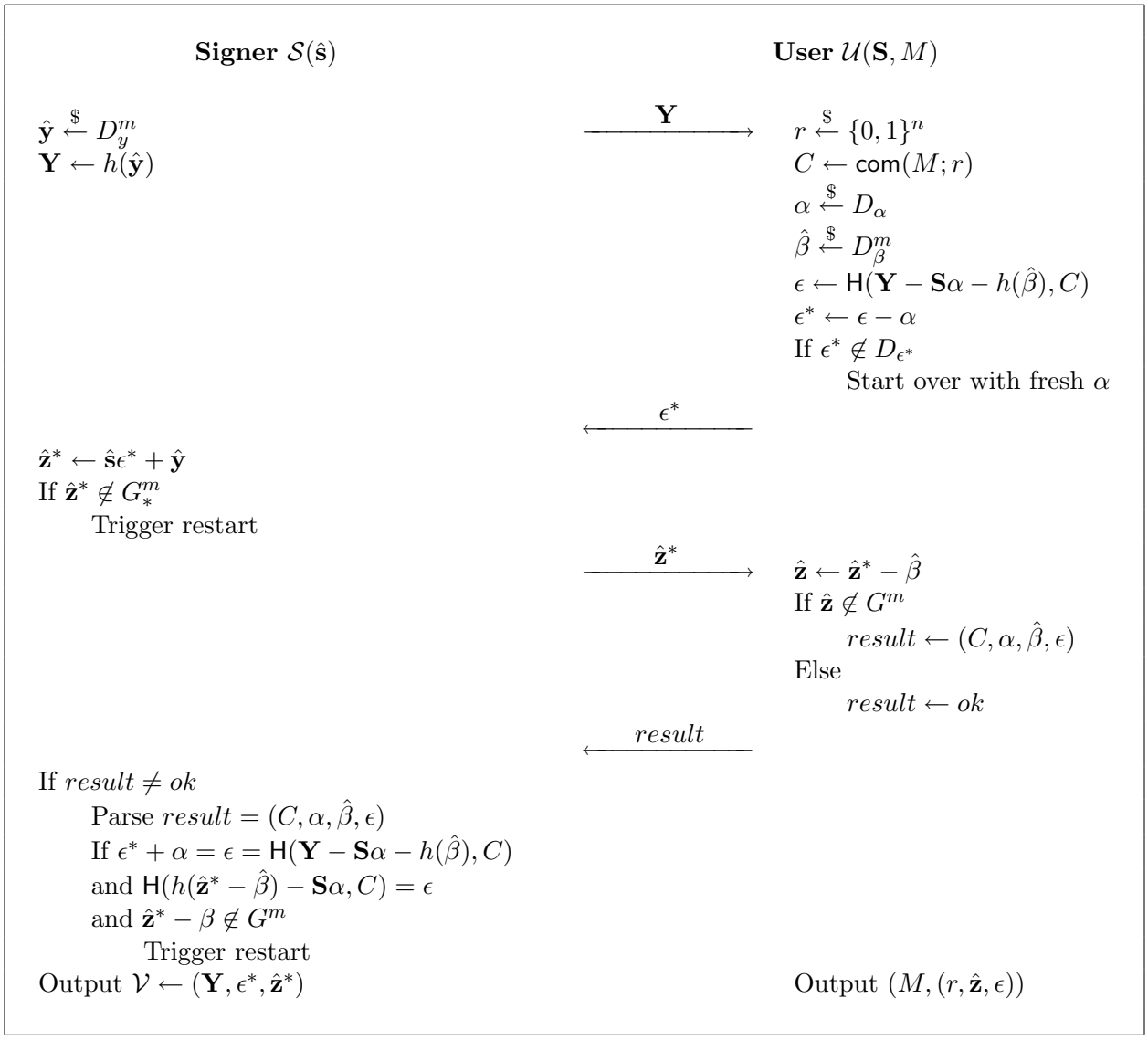

Figure 2: Issue protocol of the blind signature scheme BS.

Completeness. Assuming that the protocol does not abort, then for all honestly generated key pairs $(\hat{\mathbf{s}}, \mathbf{S})$, all messages $M \in \{0,1\}^*$, and all signatures $(r, \hat{\mathbf{z}}, \epsilon)$ we have $\hat{\mathbf{z}} \in G^m$ and $h(\hat{\mathbf{z}}) - \mathbf{S}\epsilon = h(\hat{\mathbf{z}}^* - \hat{\beta}) - \mathbf{S}\epsilon = h(\hat{\mathbf{s}}(\epsilon - \alpha) + \hat{\mathbf{y}} - \hat{\beta}) - \mathbf{S}\epsilon = \mathbf{Y} - \mathbf{S}\alpha - h(\hat{\beta})$ and $\mathsf{com}(M; r) = C$. Therefore $\mathsf{H}(h(\hat{\mathbf{z}}) - \mathbf{S}\epsilon, \mathsf{com}(M; r)) = \epsilon$ and $\mathsf{BS.Vf}(\mathbf{s}, (r, \hat{\mathbf{z}}, \epsilon), M) = 1$.

Potentially, the protocol has to be restarted a couple of times at three stages. First, the user may have to "start over with a fresh α", which is not noticed by the signer. Applying Lemma 3.2 on $\epsilon - \alpha \in D_{\epsilon^*}$ ($k = n, A = 1, B = xn$) yields a constant probability for this event.

Second, the signer may abort in case $\hat{\mathbf{z}}^* \notin G_*^m$ in order to hide its secret key. The probability for not aborting here is again constant by Lemma 3.2 ($k = mn, A = \sqrt{n}\log(n)(xn-1), B =$

$\sqrt{n}\log(n)(x^2n^2m - xnm))$ because $\|\hat{\mathbf{s}}\epsilon^*\|_\infty \leq \sqrt{n}\log(n)(xn-1)$ but with negligible probability for randomly chosen ϵ^* by [Lyu08b, Lemma 2.11].

Third, the user might abort if $\hat{\mathbf{z}} \notin G^m$. Again, Lemma 3.2 with $k = mn, A = \sqrt{n}\log(n)(x^2n^2m - xnm - xn + 1)$, and $B = \sqrt{n}\log(n)(x^3n^3m^2 - x^2n^2m^2 - x^2n^2m + xnm)$ provides that it will not abort with constant probability.

Thus, we only need a logarithmic (in n) number of trials to pass each of these aborts. In practice, 2 trials are sufficient for each of them and by choosing $x \geq 3$ we expect the protocol to be complete in a single run.

Observe that all operations in BS have $\widetilde{\mathcal{O}}(n)$ complexity and that private keys, public keys, and signatures have size $\widetilde{\mathcal{O}}(n)$.

Blindness. We prove that our scheme is $(\infty, 0)$-blind based on the observation that the signer only sees values that are statistically independent of the message being signed. More precisely, the views generated by two different messages are indistinguishable.

Theorem 4.2 (Blindness) *The blind signature scheme* BS *is* $(\infty, 0)$*-blind.*

Proof. Assuming that the signer can even control M_0, M_1 in the blindness experiment, we show that the users involved do not leak any information about their respective message whatsoever. Technically, we establish that $\epsilon^*, r, \hat{\mathbf{z}}$, and ϵ (interpreted as random variables) are distributed independently of the message being signed. As for ϵ and r we need not worry as they are chosen uniformly at random. Moreover, we need not worry about aborts in the protocol as the user makes individual runs of the protocol completely independent by choosing fresh values for $r, \alpha, \hat{\beta}$ and it never reveals r for a commitment C that was used in an aborted run. Since com is unconditionally hiding, the revealed commitment does not leak any information about the message.

Distribution of ϵ^*. With $\epsilon^*_b, \epsilon^*_{1-b}$ denote the first protocol message of $\mathcal{U}(\mathsf{pk}, M_b)$ resp. $\mathcal{U}(\mathsf{pk}, M_{1-b})$. We enforce that they are in D_{ϵ^*} and they are both of the form $\epsilon - \alpha$ with $\epsilon \in D_\epsilon$ and $\alpha \in D_\alpha$. The statistical $\Delta(\epsilon^*_b, \epsilon^*_{1-b})$ is 0 by Lemma 3.4 with $k = n, A = 1, B = xn$ because the coefficients in D_{ϵ^*} are bounded by $B - A = xn - 1$.

Distribution of $\hat{\mathbf{z}}$. With $\hat{\mathbf{z}}_0, \hat{\mathbf{z}}_1$ we denote parts of the final output of $\mathcal{U}(\mathsf{pk}, M_0)$ resp. $\mathcal{U}(\mathsf{pk}, M_1)$. Both are of the form $\hat{\mathbf{z}}^* - \hat{\beta}$ for $\hat{\mathbf{z}}^* \in G^m_*$ and a randomly chosen $\hat{\beta} \in D^m_\beta$. Forthermore, $\hat{\mathbf{z}}_0$ and $\hat{\mathbf{z}}_1$ are forced to be in G^m. Hence, their statistical distance $\Delta(\hat{\mathbf{z}}_0, \hat{\mathbf{z}}_1)$ is 0 because of Lemma 3.4 with $k = mn, A = \sqrt{n}\log(n)(x^2n^2m - xnm - xn + 1), B = \sqrt{n}\log(n)(x^3n^3m^2 - x^2n^2m^2 - x^2n^2m + xnm)$ and the fact that their coefficients are bounded by $B - A$

$$\begin{aligned} &= \sqrt{n}\log(n)(x^3n^3m^2 - x^2n^2m^2 - x^2n^2m + xnm - x^2n^2m + xnm + xn - 1) \\ &= \sqrt{n}\log(n)(x^3n^3m^2 - x^2n^2m^2 - 2x^2n^2m + 2xnm + xn - 1), \end{aligned}$$

which is exactly the norm bound enforced by "$\in G^m$". □

One-more unforgeability. BS is one-more unforgeable if there is at least one ideal lattice, corresponding to an ideal in $\mathbf{R}$, within which the ISVP^∞ is hard. We will use the forking lemma [PS00, BN06] to obtain a solution to the collision problem $Col(h, D)$, which can be used to find

short lattice vectors in the worst case via Theorem 4.1. $Col(h, D)$ is (t, δ)-hard if no t-time adversary can solve it with probability at least δ. For our proof, it is crucial that the blind signature scheme is witness-indistinguishable with respect to the private key $\hat{\mathbf{s}}$, i.e., there are at least two distinct $\hat{\mathbf{s}}, \hat{\mathbf{s}}' \in D_s$ with $h(\hat{\mathbf{s}}) = h(\hat{\mathbf{s}}')$ such that no efficient algorithm can distinguish whether $\hat{\mathbf{s}}$ or $\hat{\mathbf{s}}'$ was used by the signer with probability more than $1/2 + 2^{-\omega(\log(n))}$. We then use the forking lemma and "hope" that the adversary in the one-more unforgeability experiment outputs a signature that corresponds to a private key $\hat{\mathbf{s}}'$, while we use $\hat{\mathbf{s}}$ in the simulation. Our scheme is witness-indistinguishable because it yields valid signatures for the witness-indistinguishable signature scheme in [Lyu08b].

Theorem 4.3 (One-more unforgeability) *Let T_{Sig} and T_{H} be the cost functions for simulating the oracles* Sig *and* H*, respectively, and let $c < 1$ be the constant probability with which one protocol run has to be aborted.* BS *is $(t, q_{\mathsf{Sign}}, q_{\mathsf{H}}, \delta)$-one-more unforgeable if $Col(h, D)$ is (t', δ')-hard with $t' = t + q_{\mathsf{Sig}}T_{\mathsf{Sig}} + q_{\mathsf{H}}T_{\mathsf{H}}$ and non-negligible δ' if and only if δ is non-negligible.*

Proof. Towards contradiction, we assume that there exists a successful forger $\mathcal{A}$ against one-more unforgeability of BS. Using $\mathcal{A}$, we construct an algorithm $\mathcal{B}$ via a black-box simulation, such that $\mathcal{B}$ solves the collision problem $Col(h, D)$.

Setup. $\mathcal{B}$ gets as input the public description of h. $\mathcal{B}$ initializes a list $L_{\mathsf{H}} \leftarrow \emptyset$ of query-hash pairs $(\mathbf{R} \times \{0,1\}^*, D_\epsilon)$. It chooses $\hat{\mathbf{s}} \xleftarrow{\$} D_s$ and sets $\mathbf{S} \leftarrow h(\hat{\mathbf{s}})$. Furthermore, it randomly preselects random oracle answers $\mathbf{h}_1, \ldots, \mathbf{h}_{q_{\mathsf{H}}} \xleftarrow{\$} D_\epsilon$ and a random tape ρ. It runs $\mathcal{A}$ on input $((h, \mathbf{S}), \mathbf{h}_1, \ldots, \mathbf{h}_{q_{\mathsf{H}}}; \rho)$ in a black-box simulation.

Random oracle queries. On input $(\mathbf{u}, C)$, $\mathcal{B}$ looks up $(\mathbf{u}, C)$ in L_{H}. If it finds corresponding hash value ϵ then it returns ϵ. Otherwise, $\mathcal{B}$ chooses the first unused value ϵ from the list $\mathbf{h}_1, \ldots, \mathbf{h}_{q_{\mathsf{H}}}$, stores $((\mathbf{u}, C), \epsilon)$ in L_{H}, and returns ϵ.

Blind signature queries. On input ϵ^*, $\mathcal{B}$ acts according to BS.Sign.

Output. Finally, $\mathcal{A}$ stops and outputs $(M_1, (r_1, \hat{\mathbf{z}}_1, \epsilon_1)), \ldots, (M_\jmath, (r_\jmath, \hat{\mathbf{z}}_\jmath, \epsilon_\jmath))$, $q_{\mathsf{Sign}} < \jmath$, for distinct messages. W.l.o.g., assume that $(h(\hat{\mathbf{z}}_i) - \mathbf{S}\epsilon_i, \mathsf{com}(M_i; r_i)), \epsilon_i) \in L_{\mathsf{H}}$, for all $i = 1, \ldots, \jmath$. Algorithm $\mathcal{B}$ guesses an index $k \xleftarrow{\$} \{1, \ldots, \jmath\}$ such that $h_\imath = \epsilon_k$. Then, $\mathcal{B}$ starts over, running $\mathcal{A}$ on input $((h, \mathbf{S}), \mathbf{h}_1, \ldots, \mathbf{h}_{\imath-1}, \mathbf{h}'_\imath, \ldots, \mathbf{h}_{q_{\mathsf{H}}}; \rho)$ for a fresh set $\mathbf{h}'_\imath, \ldots, \mathbf{h}_{q_{\mathsf{H}}} \xleftarrow{\$} D_\epsilon$. Note that both $\mathcal{A}$ and $\mathcal{B}$ run with the same random tape as in the first run. Among other values, $\mathcal{A}$ outputs $(M'_k, (r'_k, \hat{\mathbf{z}}'_k, \epsilon'_k))$. $\mathcal{B}$ outputs $(\hat{\mathbf{z}}_k - \hat{\mathbf{s}}\epsilon_k, \hat{\mathbf{z}}'_k - \hat{\mathbf{s}}\epsilon'_k)$ as its solution to $Col(h, D)$.

Analysis. Observe that $\mathcal{A}$'s environment is perfectly simulated. Especially, signer-side and user-side aborts happen with the same probability as in the original protocol. The probability that $\mathcal{B}$ guesses k correctly, i.e. the k-th output of $\mathcal{A}$ corresponds to a signature that was not obtained via honestly following the protocol, is at least $1/\jmath \geq 1/(q_{\mathsf{Sign}} + 1)$. Note that ϵ_k is a random oracle answer but with probability $1/|D_\epsilon|$.

Applying Lemma 2.1 (Forking Lemma) we know that with probability $\geq (1-c)(\delta - 1/|D_\epsilon|)((\delta - 1/|D_\epsilon|)/q_{\mathsf{H}} - 1/|D_\epsilon|)$, $\mathcal{A}$ is again successful in the one-more unforgeability experiment and outputs $(M'_k, (r'_k, \hat{\mathbf{z}}'_k, \epsilon'_k))$ using the same random oracle query as in the first run. Observe that the $(1-c)$

factor takes signer-side aborts into account, which happen with probability at most c. Therefore, we know that $(h(\hat{\mathbf{z}}_k - \mathbf{S}\epsilon_k), \mathsf{com}(M_k; r_k)) = (h(\hat{\mathbf{z}}'_k - \mathbf{S}\epsilon'_k), \mathsf{com}(M'_k; r'_k))$.

Now, we turn to solving the collision problem. We have to show that $\hat{\mathbf{z}}_k - \hat{\mathbf{s}}\epsilon_k \neq \hat{\mathbf{z}}'_k - \hat{\mathbf{s}}\epsilon'_k$ and $h(\hat{\mathbf{z}}_k - \hat{\mathbf{s}}\epsilon_k) = h(\hat{\mathbf{z}}'_k - \hat{\mathbf{s}}\epsilon'_k)$. The second requirement follows directly from the previous paragraph and from the fact that we know $\hat{\mathbf{s}}$. The first is tricker. Here, it is important that the protocol is witness-indistinguishable, i.e., the adversary does not recognize whether we used one of at least two possible $\hat{\mathbf{s}}, \hat{\mathbf{s}}'$ with probability greater than $1/2 + 2^{-\omega(\log(n))}$. Thus, with probability at least $1/2 - 2^{-\omega(\log(n))}$ it will output signatures that correspond to $\hat{\mathbf{s}}'$. We apply the same proof technique as in [Lyu08b] to show that either $\hat{\mathbf{z}}_k - \hat{\mathbf{s}}\epsilon_k \neq \hat{\mathbf{z}}'_k - \hat{\mathbf{s}}\epsilon'_k$ or $\hat{\mathbf{z}}_k - \hat{\mathbf{s}}'\epsilon_k \neq \hat{\mathbf{z}}'_k - \hat{\mathbf{s}}'\epsilon'_k$. Assume both are equal, we substract the equations and obtain $(\epsilon_k - \epsilon')(\hat{\mathbf{s}}' - \hat{\mathbf{s}}) = \mathbf{0}$. We know that $\epsilon_k - \epsilon' \neq \mathbf{0}$. Now, $\|(\epsilon_k - \epsilon')(\hat{\mathbf{s}}' - \hat{\mathbf{s}})\|_\infty \leq 4n < q/2$ because $\|\epsilon_k - \epsilon'\|_\infty, \|\hat{\mathbf{s}}' - \hat{\mathbf{s}}\|_\infty \leq 2$. Thus, $(\epsilon_k - \epsilon')(\hat{\mathbf{s}}' - \hat{\mathbf{s}}) = \mathbf{0}$ over $\mathbb{Z}[X]/\langle X^n + 1\rangle$ (without $\mod q$), which is an integral domain. So, we have the contradiction $\hat{\mathbf{s}}' = \hat{\mathbf{s}}$ and a collision with probability $1/2 - 2^{-\omega(\log(n))}$ because $\hat{\mathbf{z}}_k - \hat{\mathbf{s}}\epsilon_k, \hat{\mathbf{z}}'_k - \hat{\mathbf{s}}\epsilon'_k \in D$. The success probability of this strategy is at least

$$\delta_{\mathsf{frk}} = \left(\delta - \frac{1}{|D_\epsilon|}\right)\left(1/2 - 2^{-\omega(\log(n))}\right)\frac{1}{q_{\mathsf{Sign}} + 1}(1 - c)\left(\frac{\delta - 1/|D_\epsilon|}{q_{\mathsf{H}}} - \frac{1}{|D_\epsilon|}\right),$$

which is non-negligible if δ is non-negligible.

Concerning aborts, we argue that the user cannot obtain a valid signature out of an aborted interaction. In order to trigger an abort it outputs $result = (C', \alpha', \hat{\beta}', \epsilon')$ which, together with $\hat{\mathbf{z}}^*, \hat{\mathbf{y}}, \epsilon^*$, satisfies all abort criteria. Assume that it obtains a valid signature $(r, \hat{\mathbf{z}}, \hat{\epsilon})$ from this interaction. If $\epsilon = \epsilon'$, then $h(\hat{\mathbf{z}}^* - \hat{\beta}^* - \hat{\mathbf{s}}\epsilon) = h(\hat{\mathbf{z}} - \hat{\mathbf{s}}\epsilon)$ but with probability δ_{com}. If the arguments under h are equal, we have $\hat{\mathbf{z}} = \hat{\mathbf{z}}^* - \hat{\beta} \in G^m$ — a contradiction. If the arguments are distinct, we have a collision in D. If $\epsilon \neq \epsilon'$, but necessarily $\epsilon^* = \epsilon' - \alpha' = \epsilon - \alpha$ for an $\alpha \neq \alpha$, we know that $\alpha = \epsilon - \epsilon' + \alpha'$. So, the adversary has to be able to predict the output of H, which it can only achieve with probability $1/|D_\epsilon|$. The probability that we can extract a collision from a cheating user during an abort is at least

$$\delta_{\mathsf{abort}} = \delta\left(1 - \frac{1}{|D_\epsilon|}\right)(1 - \delta_{\mathsf{com}}),$$

which is non-negligible if δ is non-negligible. Thus, the overall success probability of the reduction is $\delta' = \min(\delta_{\mathsf{frk}}, \delta_{\mathsf{abort}})$. □

By Theorem 4.1, we get the following, strong security guarantees.

Corollary 4.4 BS *is one-more unforgeable if solving* ISVP^∞ *is hard in the worst case for approximation factors* $\gamma \geq 16mn\sqrt{n}\log^2(n)(x^3n^3m^2 - x^2n^2m^2 - 2x^2n^2m + 2xnm + xn) = \widetilde{\mathcal{O}}(n^4\sqrt{n})$ *in lattices that correspond to ideals in* $\mathbf{R}$.

Comparing Corollary 4.4 with Corollary 3.1 may lead to the conclusion that the scheme in Section 3 has stronger security guarantees. This, however, is not the case because it is not provably as hard to break as finding collisions under the employed trapdoor function. Corollary 3.1 is a mere indication of hardness while Corollary 4.4 is an actual reduction from worst-case lattice problems.

5 Conclusions

We have shown how to construct efficient and provably secure blind signature schemes based on worst-case lattice problems. Our first construction is comparable to RSA blind signatures, which also rely on an interactive assumption. However, our second construction is provably secure without such assumptions and directly relies on the hardness of standard lattice problems, which are conjectured to be intractable even by quantum computers and subexponential attacks. All in all, our second scheme is the preferred scheme for practical purposes as it is more efficient and has stronger security guarantees.

Acknowledgments

The author would like to thank Richard Lindner and Michael Schneider for reviewing parts of this work and he is indepted to Vadim Lyubashevsky and Dominiqe Schröder helpful comments and discussions.

References

[Abe01] Masayuki Abe. *A Secure Three-Move Blind Signature Scheme for Polynomially Many Signatures.* Advances in Cryptology — Eurocrypt 2001, Volume 2045 of Lecture Notes in Computer Science, pages 136–151. Springer-Verlag, 2001.

[Ajt96] Miklós Ajtai. *Generating Hard Instances of Lattice Problems (Extended Abstract).* Proceedings of the Annual Symposium on the Theory of Computing (STOC) 1996, Lecture Notes in Computer Science, pages 99–108. ACM Press, 1996.

[AKS01] Miklós Ajtai, Ravi Kumar, and D. Sivakumar. *A sieve algorithm for the shortest lattice vector problem.* Proceedings of the Annual Symposium on the Theory of Computing (STOC) 2001, pages 601–610. ACM Press, 2001.

[BLR08] Johannes Buchmann, Richard Lindner, and Markus Rückert. *Explicit hard instances of the shortest vector problem.* Post-Quantum Cryptography (PQCrypto) 2008, Volume 5299 of Lecture Notes in Computer Science, pages 79–94. Springer-Verlag, 2008.

[BMV08] Emmanuel Bresson, Jean Monnerat, and Damien Vergnaud. *Separation Results on the "One-More" Computational Problems.* Topics in Cryptology — Cryptographer's Track, RSA Conference (CT-RSA)2008, Lecture Notes in Computer Science, pages 71–87. Springer-Verlag, 2008.

[BN06] Mihir Bellare and Gregory Neven. *Multi-signatures in the plain public-Key model and a general forking lemma.* ACM Conference on Computer and Communications Security'06, pages 390–399. ACM Press, 2006.

[BNPS03] Mihir Bellare, Chanathip Namprempre, David Pointcheval, and Michael Semanko. *The One-More-RSA-Inversion Problems and the Security of Chaum's Blind Signature Scheme. Journal of Cryptology*, 16(3):185–215, 2003.

[Bol03] Alexandra Boldyreva. *Threshold Signatures, Multisignatures and Blind Signatures Based on the Gap-Diffie-Hellman-Group Signature Scheme*. Public-Key Cryptography (PKC) 2003, Volume 2567 of Lecture Notes in Computer Science, pages 31–46. Springer-Verlag, 2003.

[BR93] Mihir Bellare and Pil Rogaway. *Random Oracles are Practical: A Paradigm for Designing Efficient Protocols*. Proceedings of the Annual Conference on Computer and Communications Security (CCS). ACM Press, 1993.

[Cha83] David Chaum. *Blind Signatures for Untraceable Payments*. Advances in Cryptology — Crypto 1982, pages 199–203. Plemum, New York, 1983.

[CKW04] Jan Camenisch, Maciej Koprowski, and Bogdan Warinschi. *Efficient Blind Signatures Without Random Oracles*. Security in Communication Networks, Volume 3352 of Lecture Notes in Computer Science, pages 134–148. Springer-Verlag, 2004.

[CNS07] Jan Camenisch, Gregory Neven, and Abhi Shelat. *Simulatable Adaptive Oblivious Transfer*. Advances in Cryptology — Eurocrypt 2007, Lecture Notes in Computer Science, pages 573–590. Springer-Verlag, 2007.

[Din02] Irit Dinur. *Approximating SVP_∞ to within almost-polynomial factors is NP-hard*. *Theoretical Computer Science*, 285(1):55–71, 2002.

[Fis06] Marc Fischlin. *Round-Optimal Composable Blind Signatures in the Common Reference String Model*. Advances in Cryptology — Crypto 2006, Volume 4117 of Lecture Notes in Computer Science, pages 60–77. Springer-Verlag, 2006.

[GN08] Nicolas Gama and Phong Q. Nguyen. *Predicting Lattice Reduction*. Advances in Cryptology — Eurocrypt 2008, Lecture Notes in Computer Science, pages 31–51. Springer-Verlag, 2008.

[Gol04] Oded Goldreich. *The Foundations of Cryptography*, Volume 1. Cambridge University Press, 2004.

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. *Trapdoors for hard lattices and new cryptographic constructions*. Proceedings of the Annual Symposium on the Theory of Computing (STOC) 2008, Lecture Notes in Computer Science, pages 197–206. ACM Press, 2008.

[HKKL07] Carmit Hazay, Jonathan Katz, Chiu-Yuen Koo, and Yehuda Lindell. *Concurrently-Secure Blind Signatures Without Random Oracles or Setup Assumptions*. Theory of Cryptography Conference (TCC) 2007, Volume 4392 of Lecture Notes in Computer Science, pages 323–341. Springer-Verlag, 2007.

[JLO97] Ari Juels, Michael Luby, and Rafail Ostrovsky. *Security of Blind Digital Signatures*. Advances in Cryptology — Crypto 1997, Volume 1294 of Lecture Notes in Computer Science, pages 150–164. Springer-Verlag, 1997.

[Kho05] Subhash Khot. *Hardness of approximating the shortest vector problem in lattices*. *J. ACM*, 52(5):789–808, 2005.

[Kim04] Kwangjo Kim. *Lessons from internet voting during 2002 fifa worldcup korea/japan.* 2004.

[KTX08] Akinori Kawachi, Keisuke Tanaka, and Keita Xagawa. *Concurrently Secure Identification Schemes Based on the Worst-Case Hardness of Lattice Problems.* Advances in Cryptology — Asiacrypt 2008, Volume 5350 of Lecture Notes in Computer Science, pages 372–389. Springer-Verlag, 2008.

[KZ05] Aggelos Kiayias and Hong-Sheng Zhou. *Two-Round Concurrent Blind Signatures without Random Oracles.* Number 2005/435 in Cryptology eprint archive. `eprint.iacr.org`, 2005.

[LLL82] A. Lenstra, H. Lenstra, and L. Lovász. *Factoring polynomials with rational coefficients. Mathematische Annalen*, 261(4):515–534, 1982.

[LM06] Vadim Lyubashevsky and Daniele Micciancio. *Generalized Compact Knapsacks Are Collision Resistant.* Automata, Languages and Programming (ICALP) 2006, Volume 4052 of Lecture Notes in Computer Science, pages 144–155. Springer-Verlag, 2006.

[Lyu08a] Vadim Lyubashevsky. *Lattice-Based Identification Schemes Secure Under Active Attacks.* Public-Key Cryptography (PKC) 2008, Volume 4939 of Lecture Notes in Computer Science, pages 162–179. Springer-Verlag, 2008.

[Lyu08b] Vadim Lyubashevsky. *Towards Practical Lattice-Based Cryptography.* PhD thesis, 2008.

[MR07] Daniele Micciancio and Oded Regev. *Worst-Case to Average-Case Reductions Based on Gaussian Measures. SIAM Journal on Computing*, 37(1):267–302, 2007.

[NR06] P. Q. Nguyen and O. Regev. *Learning a Parallelepiped: Cryptanalysis of GGH and NTRU Signatures.* Advances in Cryptology — Eurocrypt 2006, Volume 4004 of Lecture Notes in Computer Science, pages 215–233. Springer-Verlag, 2006.

[Oka06] Tatsuaki Okamoto. *Efficient Blind and Partially Blind Signatures Without Random Oracles.* Theory of Cryptography Conference (TCC) 2006, Volume 3876 of Lecture Notes in Computer Science, pages 80–99. Springer-Verlag, 2006.

[PPV08] Omkant Pandey, Rafael Pass, and Vinod Vaikuntanathan. *Adaptive One-Way Functions and Applications.* Advances in Cryptology — Crypto 2008, Lecture Notes in Computer Science, pages 57–74. Springer-Verlag, 2008.

[PS97] David Pointcheval and Jacques Stern. *New Blind Signatures Equivalent to Factorization (extended abstract).* ACM Conference on Computer and Communications Security, pages 92–99. ACM Press, 1997.

[PS00] David Pointcheval and Jacques Stern. *Security Arguments for Digital Signatures and Blind Signatures. Journal of Cryptology*, 13(3):361–396, 2000.

[Reg07] Oded Regev. *On the Complexity of Lattice Problems with Polynomial Approximation Factors*, 2007. A survey for the LLL+25 conference.

[RHOAGZ07] Francisco Rodríguez-Henríquez, Daniel Ortiz-Arroyo, and Claudia García-Zamora. *Yet another improvement over the Mu-Varadharajan e-voting protocol. Comput. Stand. Interfaces*, 29(4):471–480, 2007.

[RR06] Oded Regev and Ricky Rosen. *Lattice problems and norm embeddings*. Proceedings of the Annual Symposium on the Theory of Computing (STOC) 2006, pages 447–456. ACM Press, 2006.

[Sho97] Peter W. Shor. *Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. SIAM Journal on Computing*, 26(5):1484–1509, 1997.

[SSTX09] Damien Stehlé, Ron Steinfeld, Keisuke Tanaka, and Keita Xagawa. *Efficient Public Key Encryption Based on Ideal Lattices*, 2009.