Signature Schemes with Efficient Protocols and Dynamic Group Signatures from Lattice Assumptions

Benoît Libert[1], San Ling[2], Fabrice Mouhartem[1], Khoa Nguyen[2], and Huaxiong Wang[2]

[1] École Normale Supérieure de Lyon, Laboratoire LIP (France)
[2] School of Physical and Mathematical Sciences, Nanyang Technological University (Singapore)

Abstract. A recent line of works – initiated by Gordon, Katz and Vaikuntanathan (Asiacrypt 2010) – gave lattice-based constructions allowing users to authenticate while remaining hidden in a crowd. Despite five years of efforts, known constructions are still limited to static sets of users, which cannot be dynamically updated. This work provides new tools enabling the design of anonymous authentication systems whereby new users can join the system at any time.

Our first contribution is a signature scheme with efficient protocols, which allows users to obtain a signature on a committed value and subsequently prove knowledge of a signature on a committed message. This construction is well-suited to the design of anonymous credentials and group signatures. It indeed provides the first lattice-based group signature supporting dynamically growing populations of users.

As a critical component of our group signature, we provide a simple joining mechanism of introducing new group members using our signature scheme. This technique is combined with zero-knowledge arguments allowing registered group members to prove knowledge of a secret short vector of which the corresponding public syndrome was certified by the group manager. These tools provide similar advantages to those of structure-preserving signatures in the realm of bilinear groups. Namely, they allow group members to generate their own public key without having to prove knowledge of the underlying secret key. This results in a two-message joining protocol supporting concurrent enrollments, which can be used in other settings such as group encryption.

Our zero-knowledge arguments are presented in a unified framework where: (i) The involved statements reduce to arguing possession of a $\{-1, 0, 1\}$-vector $\mathbf{x}$ with a particular structure and satisfying $\mathbf{P} \cdot \mathbf{x} = \mathbf{v} \bmod q$ for some public matrix $\mathbf{P}$ and vector $\mathbf{v}$; (ii) The reduced statements can be handled using permuting techniques for Stern-like protocols. Our framework can serve as a blueprint for proving many other relations in lattice-based cryptography.

Keywords. Lattice-based cryptography, anonymity, signatures with efficient protocols, dynamic group signatures, anonymous credentials.

1 Introduction

Lattice-based cryptography is currently emerging as a promising alternative to traditional public-key techniques. During the last decade, it has received a permanent interest due to its numerous advantages. Not only does it seemingly resist quantum attacks, it also provides a better asymptotic efficiency than its relatives based on conventional number theory. While enabling many advanced functionalities [41,44,45], lattice-based primitives tend to interact with zero-knowledge proofs [43] less smoothly than their counterparts in abelian groups endowed with a bilinear map (see, e.g., [18,31,38,49,2]) or groups of hidden order [6,29,30,26]. Arguably, this partially arises from the fact that lattices have far less algebraic structure than, e.g., pairing-friendly cyclic groups. It is not surprising that the most efficient zero-knowledge proofs for lattice-related languages [15] take advantage of the extra algebraic structure available in the ring setting [64]. A consequence of the scarcity of truly efficient zero-knowledge proofs in the lattice setting is that, in the context of anonymity and privacy-preserving protocols, lattice-based cryptography has undergone significantly slower development than in other areas like functional encryption [44,45]. While natural realizations of ring signatures [70] showed up promptly [52,22] after the seminal work of Gentry, Peikert and Vaikuntanathan (GPV) [42], viable constructions of lattice-based group signatures remained lacking until the work of Gordon, Katz and Vaikuntanathan [46] in 2010. Despite recent advances [57,14,66,62], privacy-preserving primitives remain substantially less practical and powerful in terms of functionalities than their siblings based on traditional number theoretic problems [6,18,38,55] for which solutions even exist outside the random oracle model [20,21,48,10]. For example, we still have no convenient realization of group signature supporting dynamic groups [13,55] or anonymous credentials [34,28].

In this paper, we address the latter two problems by first proposing a lattice-based signature with efficient protocols in the fashion of Camenisch and Lysyanskaya [30]. To ease its use in the design of dynamic group signatures, we introduce a zero-knowledge argument system that allows a user to prove knowledge of a signature on a public key for which the user knows the underlying secret key.

Related Work. Anonymous credentials were first suggested by Chaum [34] and efficiently realized by Camenisch and Lysyanskaya [28,30]. They involve one or more credential issuer(s) and a set of users who have a long-term secret key which constitutes their digital identity and pseudonyms that can be seen as commitments to their secret key. Users can dynamically obtain credentials from an issuer that only knows users' pseudonyms and obliviously certifies users' secret keys as well as (optionally) a set of attributes. Later on, users can make themselves known to verifiers under a different pseudonym and demonstrate possession of the issuer's signature on their secret key without revealing neither the signature nor the key. Anonymous credentials typically consist of a protocol whereby the user obtains the issuer's signature on a committed message, another protocol for proving that two commitments open to the same value (which allows

proving that the same secret underlies two distinct pseudonyms) and a protocol for proving possession of a secret message-signature pair.

The first efficient constructions were given by Camenisch and Lysyanskaya under the Strong RSA assumption [28,30] or using bilinear groups [31]. Other solutions were subsequently given with additional useful properties such as non-interactivity [10], delegatability [9] or support for efficient attributes [24] (see [27] and references therein). Anonymous credentials with attributes are often obtained by having the issuer obliviously sign a multi-block message $(\mathfrak{m}_1, \ldots, \mathfrak{m}_N)$, where one block is the secret key while other blocks contain public or private attributes. Note that, for the sake of keeping the scheme compatible with zero-knowledge proofs, the blocks $(\mathfrak{m}_1, \ldots, \mathfrak{m}_N)$ cannot be simply hashed before getting signed using a ordinary, single-block signature.

Group signatures are a central anonymity primitive, introduced by Chaum and van Heyst [35] in 1991, which allows members of a group managed by some authority to sign messages in the name of the entire group. At the same time, users remain accountable for the messages they sign since an opening authority can identify them if they misbehave.

Ateniese, Camenisch, Joye and Tsudik [6] provided the first scalable construction meeting the security requirements that can be intuitively expected from the primitive, although clean security notions were not available yet at that time. Bellare, Micciancio and Warinschi [11] filled this gap by providing suitable security notions for static groups, which were subsequently extended to the dynamic setting[3] by Kiayias and Yung [55] and Bellare, Shi and Zhang [13]. In these models, efficient schemes have been put forth in the random oracle model [55,38] (the ROM) and in the standard model [48,2,1].

Lattice-based group signatures were put forth for the first time by Gordon, Katz and Vaikuntanathan [46] whose solution had linear-size signatures in the number of group members. Camenisch, Neven and Rückert [32] extended [46] so as to achieve anonymity in the strongest sense. Laguillaumie *et al.* [56] decreased the signature length to be logarithmic in the number N_{gs} of group members. While asymptotically shorter, their signatures remained space-consuming as, analogously to the Boyen-Waters group signature [20], their scheme encrypts each bit of the signer's identity individually. Simpler and more efficient solutions with $\mathcal{O}(\log N)$ signature size were given by Nguyen, Zhang and Zhang [66] and Ling, Nguyen and Wang [62]. In particular, the latter scheme [62] achieves significantly smaller signatures by encrypting all bits of the signer's identity at once. Benhamouda *et al.* [14] described a hybrid group signature that simultaneously relies on lattice assumptions (in the ring setting) and discrete-logarithm-related assumptions. Recently, Libert, Ling, Nguyen and Wang [60] obtained substantial efficiency improvements via a construction based on Merkle trees which eliminates the need for GPV trapdoors [42]. For the time being, all known group signatures are designed for static groups and analyzed in the model of Bellare,

[3] By "dynamic setting", we refer to a scenario where new group members can register at any time but, analogously to [13,55], we do not consider the orthogonal problem of user revocation here.

Micciancio and Warinschi [11], where no new group member can be introduced after the setup phase. This is somewhat unfortunate given that, in most applications of group signatures (e.g., protecting the privacy of commuters in public transportation), the dynamicity property is arguably what we need. To date, it remains an important open problem to design a lattice-based system that supports dynamically growing population of users in the models of [13,55].

Our Contributions. Our first result is a lattice-based signature with efficient protocols for multi-block messages. Namely, we provide a way for a user to obtain a signature on a committed N-block message $(\mathfrak{m}_1, \ldots, \mathfrak{m}_N)$ as well as a protocol for proving possession of a valid message-signature pair. The signature and its companion protocols can serve as a building block for lattice-based anonymous credentials and can potentially find applications in other privacy-preserving protocols (e.g., [25]) based on lattice assumptions.

The main application that we consider in this paper is the design of a lattice-based group signature scheme for dynamic groups. We prove the security of our system in the random oracle model [12] under the Short Integer Solution (SIS) and Learning With Errors (LWE) assumptions. For security parameter λ and for groups of up to N_{gs} members, the scheme features public key size $\widetilde{\mathcal{O}}(\lambda^2) \cdot \log N_{\mathsf{gs}}$, user's secret key size $\widetilde{\mathcal{O}}(\lambda)$, and signature size $\widetilde{\mathcal{O}}(\lambda) \cdot \log N_{\mathsf{gs}}$. As exhibited in Table 1, our scheme achieves a level of efficiency comparable to recent proposals based on standard (i.e., non-ideal) lattices [56,66,62,60] in the static setting [11]. In particular, the cost of moving to dynamic groups is quite reasonable: while using the scheme from [62] as a building block, our construction only lengthens the signature size by a (small) constant factor.

Scheme	LLLS [56]	NZZ [66]	LNW [62]	LLNW [60]	Ours
Group PK	$\widetilde{\mathcal{O}}(\lambda^2) \cdot \log N_{\mathsf{gs}}$	$\widetilde{\mathcal{O}}(\lambda^2)$	$\widetilde{\mathcal{O}}(\lambda^2) \cdot \log N_{\mathsf{gs}}$	$\widetilde{\mathcal{O}}(\lambda^2)$	$\widetilde{\mathcal{O}}(\lambda^2) \cdot \log N_{\mathsf{gs}}$
User's SK	$\widetilde{\mathcal{O}}(\lambda^2)$	$\widetilde{\mathcal{O}}(\lambda^2)$	$\widetilde{\mathcal{O}}(\lambda)$	$\widetilde{\mathcal{O}}(\lambda) \cdot \log N_{\mathsf{gs}}$	$\widetilde{\mathcal{O}}(\lambda)$
Signature	$\widetilde{\mathcal{O}}(\lambda) \cdot \log N_{\mathsf{gs}}$	$\widetilde{\mathcal{O}}(\lambda + \log^2 N_{\mathsf{gs}})$	$\widetilde{\mathcal{O}}(\lambda) \cdot \log N_{\mathsf{gs}}$	$\widetilde{\mathcal{O}}(\lambda) \cdot \log N_{\mathsf{gs}}$	$\widetilde{\mathcal{O}}(\lambda) \cdot \log N_{\mathsf{gs}}$

Table 1. Efficiency comparison among recent lattice-based group signatures for static groups and our dynamic scheme. The evaluation is done with respect to 2 governing parameters: security parameter λ and the maximum expected group size N_{gs}. We do not include the earlier schemes [46,32] that have signature size $\widetilde{\mathcal{O}}(\lambda^2) \cdot N_{\mathsf{gs}}$.

As a stepping stone in the design of our dynamic group signature, we also develop a zero-knowledge argument system allowing a group member to prove knowledge of a secret key (made of a short Gaussian vector) and a membership certificate issued by the group manager on the corresponding public key. Analogously to structure-preserving signatures [2], our signature scheme and zero-knowledge arguments make it possible to sign public keys without hashing them while remaining oblivious of the underlying secret key. They thus enable a round-optimal dynamic joining protocol – which allows the group manager to introduce new group members by issuing a membership certificate on their public key – which does not require any proof of knowledge on behalf of the

prospective user. As a result, the interaction is minimal: only one message is sent in each direction between the prospective user and the group manager.[4] Besides being the first lattice-based group signature for dynamic groups, our scheme thus remains secure in the setting advocated by Kiayias and Yung [54], where many users want to join the system at the same time and concurrently interact with the group manager. We believe that, analogously to structure-preserving signatures [2,1], the combination of our signature scheme and zero-knowledge arguments can serve as a building blocks for other primitives, including group encryption [53] or adaptive oblivious transfer [47].

Our Techniques. Our signature scheme with efficient protocols builds on the SIS-based signature of Böhl *et al.* [16], which is itself a variant of Boyen's signature [19]. Recall that the latter scheme involves a public key containing matrices $\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell \in \mathbb{Z}_q^{n \times m}$ and signs an ℓ-bit message $\mathfrak{m} \in \{0,1\}^\ell$ by computing a short $\mathbf{v} \in \mathbb{Z}^{2m}$ such that $[\mathbf{A} \mid \mathbf{A}_0 + \sum_{j=1}^{\ell} \mathfrak{m}[i]\mathbf{A}_j] \cdot \mathbf{v} = \mathbf{0}^n \bmod q$. The variant proposed by Böhl *et al.* [16] only uses a constant number of matrices $\mathbf{A}, \mathbf{A}_0, \mathbf{A}_1 \in \mathbb{Z}_q^{n \times m}$. Each signature is associated with a single-use tag tag (which is only used in one signing query in the proof) and the public key involves an extra matrix $\mathbf{D} \in \mathbb{Z}_q^{n \times m}$ and a vector $\mathbf{u} \in \mathbb{Z}_q^n$. A message Msg is signed by first applying a chameleon hash function $\mathbf{h} = \mathsf{CMHash}(\mathsf{Msg}, \mathbf{s}) \in \{0,1\}^m$ and signing $\mathbf{h}$ by computing a short $\mathbf{v} \in \mathbb{Z}^m$ such that $[\mathbf{A} \mid \mathbf{A}_0 + \mathsf{tag} \cdot \mathbf{A}_1] \cdot \mathbf{v} = \mathbf{u} + \mathbf{D} \cdot \mathbf{h} \bmod q$.

Our scheme extends [16] – modulo the use of a larger number of matrices $(\{\mathbf{A}_j\}_{j=0}^{\ell}, \mathbf{D}, \{\mathbf{D}\}_{k=0}^{N})$ – so that an N-block message $(\mathfrak{m}_1, \ldots, \mathfrak{m}_N) \in (\{0,1\}^L)^N$, for some $L \in \mathbb{N}$, is signed by outputting a tag $\tau \in \{0,1\}^\ell$ and a short $\mathbf{v} \in \mathbb{Z}^{2m}$ such that $[\mathbf{A} \mid \mathbf{A}_0 + \sum_{j=1}^{\ell} \tau[j] \cdot \mathbf{A}_j] \cdot \mathbf{v} = \mathbf{u} + \mathbf{D} \cdot \mathsf{CMHash}(\mathfrak{m}_1, \ldots, \mathfrak{m}_N, \mathbf{s})$, where the chameleon hash function computes $\mathbf{c}_M = \mathbf{D}_0 \cdot \mathbf{s} + \sum_{k=1}^{N} \mathbf{D}_k \cdot \mathfrak{m}_k \bmod q$, for some short vector $\mathbf{s}$, before re-encoding $\mathbf{c}_M$ so as to enable multiplication by $\mathbf{D}$.

In order to obtain a signature scheme akin to the one of Camenisch and Lysyanskaya [30], our idea is to have the tag $\tau \in \{0,1\}^\ell$ play the same role as the prime exponent in Strong-RSA-based schemes [30]. In the security proof of [16], we are faced with two situations: either the adversary produces a signature on a fresh tag $\tau^\star$, or it recycles a tag $\tau^{(i)}$ used by the signing oracle for a new, un-signed message $(\mathfrak{m}_1^\star, \ldots, \mathfrak{m}_N^\star)$. In the former case, the proof can proceed as in Boyen's proof [19]. In the latter case, the reduction must guess upfront which tag $\tau^{(i^\dagger)}$ the adversary will choose to re-use and find a way to properly answer the $i^\dagger$-th signing query without using the vanished trapdoor (for other queries, the Agrawal *et al.* technique [3] applies to compute a suitable $\mathbf{v}$ using a trapdoor hidden in $\{\mathbf{A}_j\}_{j=0}^{\ell}$). Böhl *et al.* [16] solve this problem by "programming" the vector $\mathbf{u} \in \mathbb{Z}_q^n$ in a special way and achieve full security using chameleon hashing.

To adapt this idea in the context of signatures with efficient protocols, we have to overcome several difficulties. The first one is to map $\mathbf{c}_M$ back in the domain of the chameleon hash function while preserving the compatibility with

[4] Note that each signature still requires the user to prove knowledge of his secret key. However, this is not a problem in concurrent settings as the argument of knowledge is made non-interactive via the Fiat-Shamir heuristic.

zero-knowledge proofs. To solve this problem, we extend a technique used in [60] in order to build a "zero-knowledge-friendly" chameleon hash function. This function hashes $\mathsf{Msg} = (\mathfrak{m}_1, \ldots, \mathfrak{m}_N)$ by outputting the coordinate-wise binary decomposition $\mathbf{w}$ of $\mathbf{D}_0 \cdot \mathbf{s} + \sum_{k=1}^{N} \mathbf{D}_k \cdot \mathfrak{m}_k$. If we define the "powers-of-2" matrix $\mathbf{H} = \mathbf{I} \otimes \left[1 \mid 2 \mid \ldots \mid 2^{\lceil \log q \rceil}\right]$, then we can prove that $\mathbf{w} = \mathsf{CMHash}(\mathfrak{m}_1, \ldots, \mathfrak{m}_N, \mathbf{s})$ by demonstrating the knowledge of short vectors $(\mathfrak{m}_1, \ldots, \mathfrak{m}_N, \mathbf{s}, \mathbf{w})$ such that $\mathbf{H} \cdot \mathbf{w} = \mathbf{D}_0 \cdot \mathbf{s} + \sum_{k=1}^{N} \mathbf{D}_k \cdot \mathfrak{m}_k \bmod q$, which boils down to arguing knowledge of a solution to the ISIS problem [61].

The second problem is to prove knowledge of $(\tau, \mathbf{v}, \mathbf{s})$ and $(\mathfrak{m}_1, \ldots, \mathfrak{m}_N)$ satisfying $[\mathbf{A} \mid \mathbf{A}_0 + \sum_{j=1}^{\ell} \tau[j] \cdot \mathbf{A}_j] \cdot \mathbf{v} = \mathbf{u} + \mathbf{D} \cdot \mathsf{CMHash}(\mathfrak{m}_1, \ldots, \mathfrak{m}_N, \mathbf{s})$, without revealing any of the witnesses. To this end, we provide a framework for proving all the involved statement (and many other relations that naturally arise in lattice-based cryptography) as special cases. We reduce the statements to asserting that a short integer vector $\mathbf{x}$ satisfies an equation of the form $\mathbf{P} \cdot \mathbf{x} = \mathbf{v} \bmod q$, for some public matrix $\mathbf{P}$ and vector $\mathbf{v}$, and belongs to a set VALID of short vectors with a particular structure. While the small-norm property of $\mathbf{x}$ is provable using standard techniques (e.g., [63]), we argue its membership of VALID by leveraging the properties of Stern-like protocols [72,52,61]. In particular, we rely on the fact that their underlying permutations interact well with combinatorial statements pertaining to $\mathbf{x}$, especially $\mathbf{x}$ being a bitstring with a specific pattern. We believe our framework to be of independent interest as it provides a blueprint for proving many other intricate relations in a modular manner.

When we extend the scheme with a protocol for signing committed messages, we need the signer to re-randomize the user's commitment before signing the hidden messages. This is indeed necessary to provide the reduction with a backdoor allowing to correctly answer the $i^{\dagger}$-th query by "programming" the randomness of the commitment. Since we work with integers vectors, a straightforward simulation incurs a non-negligible statistical distance between the simulated distributions of re-randomization coins and the real one (which both have a discrete Gaussian distribution). Camenisch and Lysyanskaya [30] address a similar problem by choosing the signer's randomness to be exponentially larger than that of the user's commitment so as to statistically "drown" the aforementioned discrepancy. Here, the same idea would require to work with an exponentially large modulus q. Instead, we adopt a more efficient solution, inspired by Bai *et al.* [7], which is to apply an analysis based on the Rényi divergence rather than the statistical distance. In short, the Rényi divergence's properties tell us that, if some event E occurs with noticeable probability in some probability space P, so does it in a different probability space Q for which the second order divergence $R_2(P||Q)$ is sufficiently small. In our setting, $R_2(P||Q)$ is precisely polynomially bounded since the two probability spaces only diverge in one signing query.

Our dynamic group signature scheme avoids these difficulties because the group manager only signs known messages: instead of signing the user's secret key as in anonymous credentials, it creates a membership certificate by signing the user's public key. Our zero-knowledge arguments accommodate the requirements of the scheme in the following way. In the joining protocol that dynamically

introduces new group members, the user i chooses a membership secret consisting of a short discrete Gaussian vector $\mathbf{z}_i$. This user generates a public syndrome $\mathbf{v}_i = \mathbf{F} \cdot \mathbf{z}_i \mod q$, for some public matrix $\mathbf{F}$, which constitutes his public key. In order to certify $\mathbf{v}_i$, the group manager computes the coordinate-wise binary expansion $\mathsf{bin}(\mathbf{v}_i)$ of $\mathbf{v}_i$. The vector $\mathsf{bin}(\mathbf{v}_i)$ is then signed using our signature scheme. Using the resulting signature $(\tau, \mathbf{v}, \mathbf{s})$ as a membership certificate, the group member is able to sign a message by proving that: (i) He holds a valid signature $(\tau, \mathbf{v}, \mathbf{s})$ on some secret binary message $\mathsf{bin}(\mathbf{v}_i)$; (ii) The latter vector $\mathsf{bin}(\mathbf{v}_i)$ is the binary expansion of some syndrome $\mathbf{v}_i$ of which he knows a GPV pre-image $\mathbf{z}_i$. We remark that condition (ii) can be proved by providing evidence that we have $\mathbf{v}_i = \mathbf{H} \cdot \mathsf{bin}(\mathbf{v}_i) = \mathbf{F} \cdot \mathbf{z}_i \bmod q$, for some short integer vector $\mathbf{z}_i$ and some binary $\mathsf{bin}(\mathbf{v}_i)$, where $\mathbf{H}$ is the "powers-of-2" matrix. Our abstraction of Stern-like protocols [72,52,61] allows us to efficiently argue such statements. The fact that the underlying chameleon hash function smoothly interacts with Stern-like zero-knowledge arguments is the property that maintains the user's capability of efficiently proving knowledge of the underlying secret key.

Organization. In the forthcoming sections, we first provide some background in Section 2. Our signature with efficient protocols is presented in Section 3, where we also give protocols for obtaining a signature on a committed message and proving possession of a message-signature pair. Section 4 uses our signature scheme in the design of a dynamic group signature. The details of the zero-knowledge arguments used in Section 3 and Section 4 are deferred to Section 5, where we present them in a unified framework.

2 Background and Definitions

In the following, all vectors are denoted in bold lower-case letters, whereas bold upper-case letters will be used for matrices. If $\mathbf{b} \in \mathbb{R}^n$, its Euclidean norm and infinity norm will be denoted by $\|\mathbf{b}\|$ and $\|\mathbf{b}\|_\infty$, respectively. The Euclidean norm of matrix $\mathbf{B} \in \mathbb{R}^{m \times n}$ with columns $(\mathbf{b}_i)_{i \leq n}$ is denoted by $\|\mathbf{B}\| = \max_{i \leq n} \|\mathbf{b}_i\|$. If $\mathbf{B}$ is full column-rank, we let $\widetilde{\mathbf{B}}$ denote its Gram-Schmidt orthogonalization.

When S is a finite set, we denote by $U(S)$ the uniform distribution over S and by $x \hookleftarrow D$ the action of sampling x according to the distribution D.

2.1 Lattices

A (full-rank) lattice L is defined as the set of all integer linear combinations of some linearly independent basis vectors $(\mathbf{b}_i)_{i \leq n}$ belonging to some $\mathbb{R}^n$. We work with q-ary lattices, for some prime q.

Definition 1. *Let $m \geq n \geq 1$, a prime $q \geq 2$, $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and $\mathbf{u} \in \mathbb{Z}_q^n$, define $\Lambda_q(\mathbf{A}) := \{\mathbf{e} \in \mathbb{Z}^m \mid \exists \mathbf{s} \in \mathbb{Z}_q^n \ \ s.t. \ \ \mathbf{A}^T \cdot \mathbf{s} = \mathbf{e} \bmod q\}$ as well as*

$$\Lambda_q^\perp(\mathbf{A}) := \{\mathbf{e} \in \mathbb{Z}^m \mid \mathbf{A} \cdot \mathbf{e} = \mathbf{0}^n \bmod q\}, \quad \Lambda_q^{\mathbf{u}}(\mathbf{A}) := \{\mathbf{e} \in \mathbb{Z}^m \mid \mathbf{A} \cdot \mathbf{e} = \mathbf{u} \bmod q\}$$

For any $\mathbf{t} \in \Lambda_q^{\mathbf{u}}(\mathbf{A})$, $\Lambda_q^{\mathbf{u}}(\mathbf{A}) = \Lambda_q^\perp(\mathbf{A}) + \mathbf{t}$ so that $\Lambda_q^{\mathbf{u}}(\mathbf{A})$ is a shift of $\Lambda_q^\perp(\mathbf{A})$.

For a lattice L, a vector $\mathbf{c} \in \mathbb{R}^n$ and a real $\sigma > 0$, define the function $\rho_{\sigma,\mathbf{c}}(\mathbf{x}) = \exp(-\pi\|\mathbf{x}-\mathbf{c}\|^2/\sigma^2)$. The discrete Gaussian distribution of support L, parameter σ and center $\mathbf{c}$ is defined as $D_{L,\sigma,\mathbf{c}}(\mathbf{y}) = \rho_{\sigma,\mathbf{c}}(\mathbf{y})/\rho_{\sigma,\mathbf{c}}(L)$ for any $\mathbf{y} \in L$. We denote by $D_{L,\sigma}(\mathbf{y})$ the distribution centered in $\mathbf{c} = \mathbf{0}$. We will extensively use the fact that samples from $D_{L,\sigma}$ are short with overwhelming probability.

Lemma 1 ([8, Le. 1.5]). *For any lattice $L \subseteq \mathbb{R}^n$ and positive real number $\sigma > 0$, we have $\Pr_{\mathbf{b}\hookleftarrow D_{L,\sigma}}[\|\mathbf{b}\| \leq \sqrt{n}\sigma] \geq 1 - 2^{-\Omega(n)}$.*

As shown by Gentry *et al.* [42], Gaussian distributions with lattice support can be sampled efficiently given a sufficiently short basis of the lattice.

Lemma 2 ([23, Le. 2.3]). *There exists a* PPT *(probabilistic polynomial-time) algorithm* GPVSample *that takes as inputs a basis $\mathbf{B}$ of a lattice $L \subseteq \mathbb{Z}^n$ and a rational $\sigma \geq \|\widetilde{\mathbf{B}}\| \cdot \Omega(\sqrt{\log n})$, and outputs vectors $\mathbf{b} \in L$ with distribution $D_{L,\sigma}$.*

Lemma 3 ([4, Th. 3.2]). *There exists a* PPT *algorithm* TrapGen *that takes as inputs 1^n, 1^m and an integer $q \geq 2$ with $m \geq \Omega(n \log q)$, and outputs a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and a basis $\mathbf{T_A}$ of $\Lambda_q^\perp(\mathbf{A})$ such that $\mathbf{A}$ is within statistical distance $2^{-\Omega(n)}$ to $U(\mathbb{Z}_q^{n\times m})$, and $\|\widetilde{\mathbf{T_A}}\| \leq \mathcal{O}(\sqrt{n \log q})$.*

Lemma 3 is often combined with the sampler from Lemma 2. Micciancio and Peikert [65] recently proposed a more efficient approach for this combined task, which should be preferred in practice but, for the sake of simplicity, we present our schemes using TrapGen.

We also make use of an algorithm that extends a trapdoor for $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ to a trapdoor of any $\mathbf{B} \in \mathbb{Z}_q^{n\times m'}$ whose left $n \times m$ submatrix is $\mathbf{A}$.

Lemma 4 ([33, Le. 3.2]). *There exists a* PPT *algorithm* ExtBasis *that takes as inputs a matrix $\mathbf{B} \in \mathbb{Z}_q^{n\times m'}$ whose first m columns span $\mathbb{Z}_q^n$, and a basis $\mathbf{T_A}$ of $\Lambda_q^\perp(\mathbf{A})$ where $\mathbf{A}$ is the left $n \times m$ submatrix of $\mathbf{B}$, and outputs a basis $\mathbf{T_B}$ of $\Lambda_q^\perp(\mathbf{B})$ with $\|\widetilde{\mathbf{T_B}}\| \leq \|\widetilde{\mathbf{T_A}}\|$.*

In our security proofs, analogously to [19,16] we also use a technique due to Agrawal, Boneh and Boyen [3] that implements an all-but-one trapdoor mechanism (akin to the one of Boneh and Boyen [17]) in the lattice setting.

Lemma 5 ([3, Th. 19]). *There exists a* PPT *algorithm* SampleRight *that takes as inputs matrices $\mathbf{A}, \mathbf{C} \in \mathbb{Z}_q^{n\times m}$, a low-norm matrix $\mathbf{R} \in \mathbb{Z}^{m\times m}$, a short basis $\mathbf{T_C} \in \mathbb{Z}^{m\times m}$ of $\Lambda_q^\perp(\mathbf{C})$, a vector $\mathbf{u} \in \mathbb{Z}_q^n$ and a rational σ such that $\sigma \geq \|\widetilde{\mathbf{T_C}}\| \cdot \Omega(\sqrt{\log n})$, and outputs a short vector $\mathbf{b} \in \mathbb{Z}^{2m}$ such that $\left[\mathbf{A} \mid \mathbf{A}\cdot\mathbf{R} + \mathbf{C}\right] \cdot \mathbf{b} = \mathbf{u} \bmod q$ and with distribution statistically close to $D_{L,\sigma}$ where L denotes the shifted lattice $\Lambda_q^{\mathbf{u}}\left(\left[\mathbf{A} \mid \mathbf{A}\cdot\mathbf{R} + \mathbf{C}\right]\right)$.*

2.2 Computational Problems

The security of our schemes provably relies (in the ROM) on the assumption that both algorithmic problems below are hard, i.e., cannot be solved in polynomial time with non-negligible probability and non-negligible advantage, respectively.

Definition 2. *Let m, q, β be functions of $n \in \mathbb{N}$. The Short Integer Solution problem $\mathsf{SIS}_{n,m,q,\beta}$ is, given $\mathbf{A} \hookleftarrow U(\mathbb{Z}_q^{n\times m})$, find $\mathbf{x} \in \Lambda_q^{\perp}(\mathbf{A})$ with $0 < \|\mathbf{x}\| \leq \beta$.*

If $q \geq \sqrt{n}\beta$ and $m, \beta \leq \mathsf{poly}(n)$, then $\mathsf{SIS}_{n,m,q,\beta}$ is at least as hard as standard worst-case lattice problem SIVP_γ with $\gamma = \widetilde{\mathcal{O}}(\beta\sqrt{n})$ (see, e.g., [42, Se. 9]).

Definition 3. *Let $n, m \geq 1$, $q \geq 2$, and let χ be a probability distribution on $\mathbb{Z}$. For $\mathbf{s} \in \mathbb{Z}_q^n$, let $A_{\mathbf{s},\chi}$ be the distribution obtained by sampling $\mathbf{a} \hookleftarrow U(\mathbb{Z}_q^n)$ and $e \hookleftarrow \chi$, and outputting $(\mathbf{a}, \mathbf{a}^T \cdot \mathbf{s} + e) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$. The Learning With Errors problem $\mathsf{LWE}_{n,q,\chi}$ asks to distinguish m samples chosen according to $\mathcal{A}_{\mathbf{s},\chi}$ (for $\mathbf{s} \hookleftarrow U(\mathbb{Z}_q^n)$) and m samples chosen according to $U(\mathbb{Z}_q^n \times \mathbb{Z}_q)$.*

If q is a prime power, $B \geq \sqrt{n}\omega(\log n)$, $\gamma = \widetilde{\mathcal{O}}(nq/B)$, then there exists an efficient sampleable B-bounded distribution χ (i.e., χ outputs samples with norm at most B with overwhelming probability) such that $\mathsf{LWE}_{n,q,\chi}$ is as least as hard as SIVP_γ (see, e.g., [69,68,23]).

3 A Lattice-Based Signature with Efficient Protocols

Our scheme can be seen as a variant of the Böhl *et al.* signature [16], where each signature is a triple $(\tau, \mathbf{v}, \mathbf{s})$, made of a tag $\tau \in \{0,1\}^\ell$ and integer vectors $(\mathbf{v}, \mathbf{s})$ satisfying $[\mathbf{A} \mid \mathbf{A}_0 + \sum_{j=1}^{\ell} \tau[j] \cdot \mathbf{A}_j] \cdot \mathbf{v} = \mathbf{u} + \mathbf{D} \cdot \mathbf{h} \bmod q$, where matrices $\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{D} \in \mathbb{Z}_q^{n\times m}$ are public random matrices and $\mathbf{h} \in \{0,1\}^m$ is a chameleon hash of the message which is computed using randomness $\mathbf{s}$. A difference is that, while [16] uses a short single-use tag $\tau \in \mathbb{Z}_q$, we need the tag to be an ℓ-bit string $\tau \in \{0,1\}^\ell$ which will assume the same role as the prime exponent of Camenisch-Lysyanskaya signatures [30] in the security proof.

We show that a suitable chameleon hash function makes the scheme compatible with Stern-like zero-knowledge arguments [61,62] for arguing possession of a valid message-signature pair. Section 5 shows how to translate such a statement into asserting that a short witness vector $\mathbf{x}$ with a particular structure satisfies a relation of the form $\mathbf{P} \cdot \mathbf{x} = \mathbf{v} \bmod q$, for some public matrix $\mathbf{P}$ and vector $\mathbf{v}$. The underlying chameleon hash can be seen as a composition of the chameleon hash of [33, Section 4.1] with a technique used in [67,60]: on input of a message $(\mathfrak{m}_1, \ldots, \mathfrak{m}_N)$, it outputs the binary decomposition of $\mathbf{D}_0 \cdot \mathbf{s} + \sum_{k=1}^{N} \mathbf{D}_k \cdot \mathfrak{m}_k$, for some discrete Gaussian vector $\mathbf{s}$.

3.1 Description

We assume that messages are vectors of N blocks $\mathsf{Msg} = (\mathfrak{m}_1, \ldots, \mathfrak{m}_N)$, where each block is a $2m$-bit string $\mathfrak{m}_k = \mathfrak{m}_k[1] \ldots \mathfrak{m}_k[2m] \in \{0,1\}^{2m}$ for $k \in \{1, \ldots, N\}$.

For each vector $\mathbf{v} \in \mathbb{Z}_q^L$, we denote by $\mathsf{bin}(\mathbf{v}) \in \{0,1\}^{L\lceil \log q \rceil}$ the vector obtained by replacing each coordinate of $\mathbf{v}$ by its binary representation.

Keygen$(1^\lambda, 1^N)$**:** Given a security parameter $\lambda > 0$ and the number of blocks $N = \mathsf{poly}(\lambda)$, choose the following parameters: $n = \mathcal{O}(\lambda)$; a prime modulus $q = \widetilde{\mathcal{O}}(N \cdot n^4)$; dimension $m = 2n\lceil \log q \rceil$; an integer $\ell = \Theta(\lambda)$; and Gaussian parameters $\sigma = \Omega(\sqrt{n \log q} \log n)$, $\sigma_0 = 2\sqrt{2}(N+1)\sigma m^{3/2}$, and $\sigma_1 = \sqrt{\sigma_0^2 + \sigma^2}$. Define the message space as $(\{0,1\}^{2m})^N$.

1. Run $\mathsf{TrapGen}(1^n, 1^m, q)$ to get $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and a short basis $\mathbf{T_A}$ of $\Lambda_q^\perp(\mathbf{A})$. This basis allows computing short vectors in $\Lambda_q^\perp(\mathbf{A})$ with a Gaussian parameter σ. Next, choose $\ell + 1$ random $\mathbf{A}_0, \mathbf{A}_1, \ldots, \mathbf{A}_\ell \hookleftarrow U(\mathbb{Z}_q^{n \times m})$.
2. Choose random matrices $\mathbf{D} \hookleftarrow U(\mathbb{Z}_q^{n \times m})$, $\mathbf{D}_0, \mathbf{D}_1, \ldots, \mathbf{D}_N \hookleftarrow U(\mathbb{Z}_q^{2n \times 2m})$ as well as a random vector $\mathbf{u} \hookleftarrow U(\mathbb{Z}_q^n)$.

The private key consists of $SK := \mathbf{T_A} \in \mathbb{Z}^{m \times m}$ and the public key is

$$PK := \big(\mathbf{A},\ \{\mathbf{A}_j\}_{j=0}^{\ell},\ \{\mathbf{D}_k\}_{k=0}^{N},\ \mathbf{D},\ \mathbf{u}\big).$$

Sign$\big(SK, \mathsf{Msg}\big)$**:** To sign an N-block message $\mathsf{Msg} = (\mathfrak{m}_1, \ldots, \mathfrak{m}_N) \in \big(\{0,1\}^{2m}\big)^N$,

1. Choose a random string $\tau \hookleftarrow U(\{0,1\}^\ell)$. Then, using $SK := \mathbf{T_A}$, compute with $\mathsf{ExtBasis}$ a short delegated basis $\mathbf{T}_\tau \in \mathbb{Z}^{2m \times 2m}$ for the matrix

$$\mathbf{A}_\tau = [\mathbf{A} \mid \mathbf{A}_0 + \sum_{j=1}^{\ell} \tau[j]\mathbf{A}_j] \in \mathbb{Z}_q^{n \times 2m}. \tag{1}$$

2. Sample a vector $\mathbf{s} \hookleftarrow D_{\mathbb{Z}^{2m}, \sigma_1}$. Compute $\mathbf{c}_M \in \mathbb{Z}_q^{2n}$ as a chameleon hash of $(\mathfrak{m}_1, \ldots, \mathfrak{m}_N)$: i.e., compute $\mathbf{c}_M = \mathbf{D}_0 \cdot \mathbf{s} + \sum_{k=1}^{N} \mathbf{D}_k \cdot \mathfrak{m}_k \in \mathbb{Z}_q^{2n}$, which is used to define $\mathbf{u}_M = \mathbf{u} + \mathbf{D} \cdot \mathsf{bin}(\mathbf{c}_M) \in \mathbb{Z}_q^n$. Then, using the delegated basis $\mathbf{T}_\tau \in \mathbb{Z}^{2m \times 2m}$, sample a short vector $\mathbf{v} \in \mathbb{Z}^{2m}$ in $D_{\Lambda_q^{\mathbf{u}_M}(\mathbf{A}_\tau), \sigma}$.

Output the signature $sig = (\tau, \mathbf{v}, \mathbf{s}) \in \{0,1\}^\ell \times \mathbb{Z}^{2m} \times \mathbb{Z}^{2m}$.

Verify$\big(PK, \mathsf{Msg}, sig\big)$**:** Given PK, $\mathsf{Msg} = (\mathfrak{m}_1, \ldots, \mathfrak{m}_N) \in (\{0,1\}^{2m})^N$ and $sig = (\tau, \mathbf{v}, \mathbf{s}) \in \{0,1\}^\ell \times \mathbb{Z}^{2m} \times \mathbb{Z}^{2m}$, return 1 if $\|\mathbf{v}\| < \sigma\sqrt{2m}$, $\|\mathbf{s}\| < \sigma_1\sqrt{2m}$ and

$$\mathbf{A}_\tau \cdot \mathbf{v} = \mathbf{u} + \mathbf{D} \cdot \mathsf{bin}(\mathbf{D}_0 \cdot \mathbf{s} + \sum_{k=1}^{N} \mathbf{D}_k \cdot \mathfrak{m}_k) \bmod q. \tag{2}$$

When the scheme is used for obliviously signing committed messages, the security proof follows Bai *et al.* [7] in that it applies an argument based on the Rényi divergence in one signing query. This argument requires to sample $\mathbf{s}$ from a Gaussian distribution whose standard deviation σ_1 is polynomially larger than σ.

We note that, instead of being included in the public key, the matrices $\{\mathbf{D}_k\}_{k=0}^N$ can be part of public parameters shared by many signers. Indeed, only the matrices $(\mathbf{A}, \{\mathbf{A}_i\}_{i=0}^\ell)$ should be specific to the user who holds $SK = \mathbf{T_A}$. In Section 3.3, we use a variant where $\{\mathbf{D}_k\}_{k=0}^N$ belong to public parameters.

3.2 Security Analysis

The security analysis in Theorem 1 requires that $q > \ell$.

Theorem 1. *The signature scheme is secure under chosen-message attacks under the* SIS *assumption.*

Proof (Sketched). To prove the result, we will distinguish three kinds of attacks:

Type I attacks are attacks where, in the adversary's forgery $sig^\star = (\tau^\star, \mathbf{v}^\star, \mathbf{s}^\star)$, $\tau^\star$ did not appear in any output of the signing oracle.

Type II attacks are such that, in the adversary's forgery $sig^\star = (\tau^\star, \mathbf{v}^\star, \mathbf{s}^\star)$, $\tau^\star$ is recycled from an output $sig^{(i^\star)} = (\tau^{(i^\star)}, \mathbf{v}^{(i^\star)}, \mathbf{s}^{(i^\star)})$ of the signing oracle, for some index $i^\star \in \{1, \ldots, Q\}$. However, if $\mathsf{Msg}^\star = (\mathfrak{m}_1^\star, \ldots, \mathfrak{m}_N^\star)$ and $\mathsf{Msg}^{(i^\star)} = (\mathfrak{m}_1^{(i^\star)}, \ldots, \mathfrak{m}_N^{(i^\star)})$ denote the forgery message and the $i^\star$-th signing query, respectively, we have $\mathbf{D}_0 \cdot \mathbf{s}^\star + \sum_{k=1}^N \mathbf{D}_k \cdot \mathfrak{m}_k^\star \neq \mathbf{D}_0 \cdot \mathbf{s}^{(i^\star)} + \sum_{k=1}^N \mathbf{D}_k \cdot \mathfrak{m}_k^{(i^\star)}$.

Type III attacks are those where the adversary's forgery $sig^\star = (\tau^\star, \mathbf{v}^\star, \mathbf{s}^\star)$ recycles $\tau^\star$ from an output $sig^{(i^\star)} = (\tau^{(i^\star)}, \mathbf{v}^{(i^\star)}, \mathbf{s}^{(i^\star)})$ of the signing oracle (i.e., $\tau^{(i^\star)} = \tau^\star$ for some index $i^\star \in \{1, \ldots, Q\}$) and we have the collision

$$\mathbf{D}_0 \cdot \mathbf{s}^\star + \sum_{k=1}^N \mathbf{D}_k \cdot \mathfrak{m}_k^\star = \mathbf{D}_0 \cdot \mathbf{s}^{(i^\star)} + \sum_{k=1}^N \mathbf{D}_k \cdot \mathfrak{m}_k^{(i^\star)}. \tag{3}$$

Type III attacks imply a collision for the chameleon hash function of Kawachi *et al.* [52]: if (3) holds, a short vector of $\Lambda_q^\perp([\mathbf{D}_0 \mid \mathbf{D}_1 \mid \ldots \mid \mathbf{D}_N])$ is obtained as so that a collision breaks the SIS assumption.

The security against Type I attacks is proved by Lemma 6 which applies the same technique as in [19,65]. In particular, the prefix guessing technique of [50] allows keeping the modulus smaller than the number Q of adversarial queries as in [65]. In order to deal with Type II attacks, we can leverage the technique of [16]. In Lemma 7, we prove that Type II attack would also contradict SIS. □

The following lemmas are proved in the full version of the paper [59].

Lemma 6. *The scheme is secure against Type I attacks if the* $\mathsf{SIS}_{n,m,q,\beta'}$ *assumption holds for* $\beta' = m^{3/2}\sigma^2(\ell + 3) + m^{1/2}\sigma_1$.

Lemma 7. *The scheme is secure against Type II attacks under the* $\mathsf{SIS}_{n,m,q,\beta''}$ *assumption for* $\beta'' = \sqrt{2}(\ell + 2)\sigma^2 m^{3/2} + m^{1/2}$.

3.3 Protocols for Signing a Committed Value and Proving Possession of a Signature

We first show a two-party protocol whereby a user can interact with the signer in order to obtain a signature on a committed message.

In order to prove that the scheme still guarantees unforgeability for obliviously signed messages, we will assume that each message block $\mathfrak{m}_k \in \{0,1\}^{2m}$ is

obtained by encoding the actual message $M_k = M_k[1] \dots M_k[m] \in \{0,1\}^m$ as $\mathfrak{m}_k = \mathsf{Encode}(M_k) = (\bar{M}_k[1], M_k[1], \dots, \bar{M}_k[m], M_k[m])$. Namely, each 0 (resp. each 1) is encoded as a pair $(1,0)$ (resp. $(0,1)$). The reason for this encoding is that the proof of Theorem 2 requires that at least one block $\mathfrak{m}_k^\star$ of the forgery message is 1 while the same bit is 0 at some specific signing query. We will show (see Section 5) that the correctness of this encoding can be efficiently proved using Stern-like [72] protocols.

To sign committed messages, a first idea is exploit the fact that our signature of Section 3.1 blends well with the SIS-based commitment scheme suggested by Kawachi *et al.* [52]. In the latter scheme, the commitment key consists of matrices $(\mathbf{D}_0, \mathbf{D}_1) \in \mathbb{Z}_q^{2n\times 2m} \times \mathbb{Z}_q^{2n\times 2m}$, so that message $\mathfrak{m} \in \{0,1\}^{2m}$ can be committed to by sampling a Gaussian vector $\mathbf{s} \hookleftarrow D_{\mathbb{Z}^{2m},\sigma}$ and computing $\mathbf{C} = \mathbf{D}_0 \cdot \mathbf{s} + \mathbf{D}_1 \cdot \mathfrak{m} \in \mathbb{Z}_q^{2n}$. This scheme extends to commit to multiple messages $(\mathfrak{m}_1, \dots, \mathfrak{m}_N)$ at once by computing $\mathbf{C} = \mathbf{D}_0 \cdot \mathbf{s} + \sum_{k=1}^{N} \mathbf{D}_k \cdot \mathfrak{m}_k \in \mathbb{Z}_q^{2n}$ using a longer commitment key $(\mathbf{D}_0, \mathbf{D}_1, \dots, \mathbf{D}_N) \in (\mathbb{Z}_q^{2n\times 2m})^{N+1}$. It is easy to see that the resulting commitment remains statistically hiding and computationally binding under the SIS assumption.

In order to make our construction usable in the definitional framework of Camenisch *et al.* [27], we assume common public parameters (i.e., a common reference string) and encrypt all witnesses of which knowledge is being proved under a public key included in the common reference string. The resulting ciphertexts thus serve as statistically binding commitments to the witnesses. To enable this, the common public parameters comprise public keys $\mathbf{G}_0 \in \mathbb{Z}_q^{n\times \ell}$, $\mathbf{G}_1 \in \mathbb{Z}_q^{n\times 2m}$ for multi-bit variants of the dual Regev cryptosystem [42] and all parties are denied access to the underlying private keys. The flexibility of Stern-like protocols allows us to prove that the content of a perfectly hiding commitment $\mathbf{c}_\mathfrak{m}$ is consistent with encrypted values.

Global-Setup: Let $B = \sqrt{n}\omega(\log n)$ and let χ be a B-bounded distribution. Let $p = \sigma \cdot \omega(\sqrt{m})$ upper-bound entries of vectors sampled from the distribution $D_{\mathbb{Z}^{2m},\sigma}$. Generate two public keys for the dual Regev encryption scheme in its multi-bit variant. These keys consists of a public random matrix $\mathbf{B} \hookleftarrow U(\mathbb{Z}_q^{n\times m})$ and random matrices $\mathbf{G}_0 = \mathbf{B} \cdot \mathbf{E}_0 \in \mathbb{Z}_q^{n\times \ell}$, $\mathbf{G}_1 = \mathbf{B} \cdot \mathbf{E}_1 \in \mathbb{Z}_q^{n\times 2m}$, where $\mathbf{E}_0 \in \mathbb{Z}^{m\times \ell}$ and $\mathbf{E}_1 \in \mathbb{Z}^{m\times 2m}$ are short Gaussian matrices with columns sampled from $D_{\mathbb{Z}^m,\sigma}$. These matrices will be used to encrypt integer vectors of dimension ℓ and $2m$, respectively. Finally, generate public parameters $CK := \{\mathbf{D}_k\}_{k=0}^{N}$ consisting of uniformly random matrices $\mathbf{D}_k \hookleftarrow U(\mathbb{Z}_q^{2n\times 2m})$ for a statistically hiding commitment to vectors in $(\{0,1\}^{2m})^N$. Return public parameters consisting of

$$\mathsf{par} := \{\ \mathbf{B} \in \mathbb{Z}_q^{n\times m},\ \mathbf{G}_0 \in \mathbb{Z}_q^{n\times \ell},\ \mathbf{G}_1 \in \mathbb{Z}_q^{n\times 2m},\ CK\}.$$

Issue $\leftrightarrow$ Obtain : The signer S, who has $PK := \{\mathbf{A},\ \{\mathbf{A}_j\}_{j=0}^{\ell},\ \mathbf{D},\ \mathbf{u}\}$ and $SK := \mathbf{T}_\mathbf{A}$, interacts with the user U, who has $(\mathfrak{m}_1, \dots, \mathfrak{m}_N)$, as follows.

1. U samples $\mathbf{s}' \hookleftarrow D_{\mathbb{Z}^{2m},\sigma}$ and computes $\mathbf{c}_\mathfrak{m} = \mathbf{D}_0 \cdot \mathbf{s}' + \sum_{k=1}^{N} \mathbf{D}_k \cdot \mathfrak{m}_k \in \mathbb{Z}_q^{2n}$ which is sent to S as a commitment to $(\mathfrak{m}_1, \dots, \mathfrak{m}_N)$. Next, U encrypts

$\{\mathfrak{m}_k\}_{k=1}^N$ and $\mathbf{s}'$ under the key $(\mathbf{B}, \mathbf{G}_1)$ by computing for all $k \in [1, N]$:

$$\begin{aligned}\mathbf{c}_k &= (\mathbf{c}_{k,1}, \mathbf{c}_{k,2}) \\ &= \big(\mathbf{B}^T \cdot \mathbf{s}_k + \mathbf{e}_{k,1},\ \mathbf{G}_1^T \cdot \mathbf{s}_k + \mathbf{e}_{k,2} + \mathfrak{m}_k \cdot \lfloor q/2 \rfloor\big) \in \mathbb{Z}_q^m \times \mathbb{Z}_q^{2m} \end{aligned} \quad (4)$$

for randomly chosen $\mathbf{s}_k \hookleftarrow \chi^n$, $\mathbf{e}_{k,1} \hookleftarrow \chi^m$, $\mathbf{e}_{k,2} \hookleftarrow \chi^{2m}$, and

$$\begin{aligned}\mathbf{c}_{s'} &= (\mathbf{c}_{s',1}, \mathbf{c}_{s',2}) \\ &= \big(\mathbf{B}^T \cdot \mathbf{s}_0 + \mathbf{e}_{0,1},\ \mathbf{G}_1^T \cdot \mathbf{s}_0 + \mathbf{e}_{0,2} + \mathbf{s}' \cdot \lfloor q/p \rfloor\big) \in \mathbb{Z}_q^m \times \mathbb{Z}_q^{2m} \end{aligned} \quad (5)$$

where $\mathbf{s}_0 \hookleftarrow \chi^n$, $\mathbf{e}_{0,1} \hookleftarrow \chi^m$, $\mathbf{e}_{0,2} \hookleftarrow \chi^{2m}$. The ciphertexts $\{\mathbf{c}_k\}_{k=1}^N$ and $\mathbf{c}_{s'}$ are sent to S along with $\mathbf{c}_{\mathfrak{m}}$.
Then, U generates an interactive zero-knowledge argument to convince S that $\mathbf{c}_{\mathfrak{m}}$ is a commitment to $(\mathfrak{m}_1, \ldots, \mathfrak{m}_N)$ with the randomness $\mathbf{s}'$ such that $\{\mathfrak{m}_k\}_{k=1}^N$ and $\mathbf{s}'$ were honestly encrypted to $\{\mathbf{c}_k\}_{i=1}^N$ and $\mathbf{c}_{s'}$, as in (4) and (5). For convenience, this argument system will be described in Section 5.3, where we demonstrate that, together with other zero-knowledge protocols used in this work, it can be derived from a Stern-like [72] protocol constructed in Section 5.1.

2. If the argument of step 1 properly verifies, S samples $\mathbf{s}'' \hookleftarrow D_{\mathbb{Z}^{2m},\sigma_0}$ and computes a vector $\mathbf{u}_{\mathfrak{m}} = \mathbf{u} + \mathbf{D} \cdot \mathsf{bin}\big(\mathbf{c}_{\mathfrak{m}} + \mathbf{D}_0 \cdot \mathbf{s}''\big) \in \mathbb{Z}_q^n$. Next, S randomly picks $\tau \hookleftarrow \{0,1\}^\ell$ and uses $\mathbf{T}_{\mathbf{A}}$ to compute a delegated basis $\mathbf{T}_\tau \in \mathbb{Z}^{2m \times 2m}$ for the matrix $\mathbf{A}_\tau \in \mathbb{Z}_q^{n \times 2m}$ of (1). Using $\mathbf{T}_\tau \in \mathbb{Z}^{2m \times 2m}$, S samples a short vector $\mathbf{v} \in \mathbb{Z}^{2m}$ in $D^{\mathbf{u}_M}_{\Lambda^\perp(\mathbf{A}_\tau),\sigma}$. It returns the vector $(\tau, \mathbf{v}, \mathbf{s}'') \in \{0,1\}^\ell \times \mathbb{Z}^{2m} \times \mathbb{Z}^{2m}$ to U.
3. U computes $\mathbf{s} = \mathbf{s}' + \mathbf{s}''$ over $\mathbb{Z}$ and verifies that

$$\mathbf{A}_\tau \cdot \mathbf{v} = \mathbf{u} + \mathbf{D} \cdot \mathsf{bin}\big(\mathbf{D}_0 \cdot \mathbf{s} + \sum_{k=1}^N \mathbf{D}_k \cdot \mathfrak{m}_k\big) \bmod q.$$

If so, it outputs $(\tau, \mathbf{v}, \mathbf{s})$. Otherwise, it outputs $\perp$.

Note that, if both parties faithfully run the protocol, the user obtains a valid signature $(\tau, \mathbf{v}, \mathbf{s})$ for which the distribution of $\mathbf{s}$ is $D_{\mathbb{Z}^{2m},\sigma_1}$, where $\sigma_1 = \sqrt{\sigma^2 + \sigma_0^2}$.

The following protocol allows proving possession of a message-signature pair.

Prove: On input of a signature $(\tau, \mathbf{v} = (\mathbf{v}_1^T \mid \mathbf{v}_2^T)^T, \mathbf{s}) \in \{0,1\}^\ell \times \mathbb{Z}^{2m} \times \mathbb{Z}^{2m}$ on the message $(\mathfrak{m}_1, \ldots, \mathfrak{m}_N)$, the user does the following.

1. Using $(\mathbf{B}, \mathbf{G}_0)$ and $(\mathbf{B}, \mathbf{G}_1)$ generate perfectly binding commitments to $\tau \in \{0,1\}^\ell$, $\{\mathfrak{m}_k\}_{k=1}^N$, $\mathbf{v}_1, \mathbf{v}_2 \in \mathbb{Z}^m$ and $\mathbf{s} \in \mathbb{Z}^{2m}$. Namely, compute

$$\begin{aligned}\mathbf{c}_\tau &= (\mathbf{c}_{\tau,1}, \mathbf{c}_{\tau,2}) \\ &= \big(\mathbf{B}^T \cdot \mathbf{s}_\tau + \mathbf{e}_{\tau,1},\ \mathbf{G}_0^T \cdot \mathbf{s}_\tau + \mathbf{e}_{\tau,2} + \tau \cdot \lfloor q/2 \rfloor\big) \in \mathbb{Z}_q^m \times \mathbb{Z}_q^\ell, \\ \mathbf{c}_k &= (\mathbf{c}_{k,1}, \mathbf{c}_{k,2}) \ \in \mathbb{Z}_q^m \times \mathbb{Z}_q^{2m} \\ &= \big(\mathbf{B}^T \cdot \mathbf{s}_k + \mathbf{e}_{k,1},\ \mathbf{G}_1^T \cdot \mathbf{s}_k + \mathbf{e}_{k,2} + \mathfrak{m}_k \cdot \lfloor q/2 \rfloor\big) \qquad \forall k \in [1, N]\end{aligned}$$

where $\mathbf{s}_\tau, \mathbf{s}_k \hookleftarrow \chi^n$, $\mathbf{e}_{\tau,1}, \mathbf{e}_{k,1} \hookleftarrow \chi^m$, $\mathbf{e}_{\tau,2} \hookleftarrow \chi^\ell$, $\mathbf{e}_{k,2} \hookleftarrow \chi^{2m}$, as well as

$$\begin{aligned}\mathbf{c}_\mathbf{v} &= (\mathbf{c}_{\mathbf{v},1}, \mathbf{c}_{\mathbf{v},2}) \\ &= \left(\mathbf{B}^T \cdot \mathbf{s}_\mathbf{v} + \mathbf{e}_{\mathbf{v},1},\ \mathbf{G}_1^T \cdot \mathbf{s}_\mathbf{v} + \mathbf{e}_{\mathbf{v},2} + \mathbf{v} \cdot \lfloor q/p \rfloor\right) \in \mathbb{Z}_q^m \times \mathbb{Z}_q^{2m} \\ \mathbf{c}_s &= (\mathbf{c}_{s,1}, \mathbf{c}_{s,2}) \\ &= \left(\mathbf{B}^T \cdot \mathbf{s}_0 + \mathbf{e}_{0,1},\ \mathbf{G}_1^T \cdot \mathbf{s}_0 + \mathbf{e}_{0,2} + \mathbf{s} \cdot \lfloor q/p \rfloor\right) \in \mathbb{Z}_q^m \times \mathbb{Z}_q^{2m},\end{aligned}$$

where $\mathbf{s}_\mathbf{v}, \mathbf{s}_0 \hookleftarrow \chi^n$, $\mathbf{e}_{\mathbf{v},1}, \mathbf{e}_{0,1} \hookleftarrow \chi^m$, $\mathbf{e}_{\mathbf{v},2}, \mathbf{e}_{0,2} \hookleftarrow \chi^{2m}$.

2. Prove in zero-knowledge that $\mathbf{c}_\tau$, $\mathbf{c}_s$, $\mathbf{c}_\mathbf{v}$, $\{\mathbf{c}_k\}_{k=1}^N$ encrypt a valid message-signature pair. In Section 5.4, we show that this involved zero-knowledge protocol can be derived from the statistical zero-knowledge argument of knowledge for a simpler, but more general relation that we explicitly present in Section 5.1. The proof system can be made statistically ZK for a malicious verifier using standard techniques (assuming a common reference string, we can use [36]). In the random oracle model, it can be made non-interactive using the Fiat-Shamir heuristic [40].

We require that the adversary be unable to prove possession of a signature of a message $(\mathfrak{m}_1, \ldots, \mathfrak{m}_N)$ for which it did not legally obtain a credential by interacting with the issuer. Note that the messages that are blindly signed by the issuer are uniquely defined since, at each signing query, the adversary is required to supply perfectly binding commitments $\{\mathbf{c}_k\}_{k=1}^N$ to $(\mathfrak{m}_1, \ldots, \mathfrak{m}_N)$.

In instantiations using non-interactive proofs, we assume that these can be bound to a verifier-chosen nonce to prevent replay attacks, as suggested in [27].

The security proof (in Theorem 2) makes crucial use of the Rényi divergence using arguments in the spirit of Bai *et al.* [7]. The reduction has to guess upfront the index $i^\star \in \{1, \ldots, Q\}$ of the specific signing query for which the adversary will re-use $\tau^{(i^\star)}$. For this query, the reduction will have to make sure that the simulation trapdoor of Agrawal *et al.* [3] (used by the $\mathsf{SampleRight}$ algorithm of Lemma 5) vanishes: otherwise, the adversary's forgery would not be usable for solving SIS. This means that, as in the proof of [16], the reduction must answer exactly one signing query in a different way, without using the trapdoor. While Böhl *et al.* solve this problem by exploiting the fact that they only need to prove security against non-adaptive forgers, we directly use a built-in chameleon hash function mechanism which is implicitly realized by the matrix $\mathbf{D}_0$ and the vector $\mathbf{s}$. Namely, in the signing query for which the Agrawal *et al.* trapdoor [3] cancels, we assign a special value to the vector $\mathbf{s} \in \mathbb{Z}^{2m}$, which depends on the adaptively-chosen signed message $(\mathsf{Msg}_1^{(i^\star)}, \ldots, \mathsf{Msg}_N^{(i^\star)})$ and some Gaussian matrices $\{\mathbf{R}_k\}_{k=1}^N$ hidden behind $\{\mathbf{D}_k\}_{k=1}^N$.

One issue is that this results in a different distribution for the vector $\mathbf{s} \in \mathbb{Z}^m$. However, we can still view $\mathbf{s}$ as a vector sampled from a Gaussian distribution centered away from $\mathbf{0}^{2m}$. Since this specific situation occurs only once during the simulation, we can apply a result proved in [58] which upper-bounds the Rényi divergence between two Gaussian distributions with identical standard deviations but different centers. By choosing the standard deviation σ_1 of $\mathbf{s} \in \mathbb{Z}^{2m}$ to be polynomially larger than that of the columns of matrices $\{\mathbf{R}_k\}_{k=1}^N$, we can keep

the Rényi divergence between the two distributions of $\mathbf{s}$ (i.e., the one of the simulation and the one of the real game) sufficiently small to apply the probability preservation property (which still gives a polynomial reduction since the argument must only be applied on one signing query). Namely, the latter implies that, if the Rényi divergence $R_2(\mathbf{s}^{\mathsf{real}}||\mathbf{s}^{\mathsf{sim}})$ is polynomial, the probability that the simulated vector $\mathbf{s}^{\mathsf{sim}} \in \mathbb{Z}^{2m}$ passes the verification test will only be polynomially smaller than in the real game and so will be the adversary's probability of success.

Another option would have been to keep the statistical distance between $\mathbf{s}^{\mathsf{real}}$ and $\mathbf{s}^{\mathsf{sim}}$ negligible using the smudging technique of [5]. However, this would have implied to use an exponentially large modulus q since σ_1 should have been exponentially larger than the standard deviations of the columns of $\{\mathbf{R}_k\}_{k=1}^{N}$.

The proofs of the following theorems are given in the full version of the paper.

Theorem 2. *Under the* $\mathsf{SIS}_{n,2m,q,\hat{\beta}}$ *assumption, where* $\hat{\beta} = N\sigma(2m)^{3/2}+4\sigma_1 m^{3/2}$*, the above protocols are secure protocols for obtaining a signature on a committed message and proving possession of a valid message-signature pair.*

Theorem 3. *The scheme provides anonymity under the* $\mathsf{LWE}_{n,q,\chi}$ *assumption.*

4 A Dynamic Lattice-Based Group Signature

In this section, the signature scheme of Section 3 is used to design a group signature for dynamic groups using the syntax and the security model of Kiayias and Yung [55], which is recalled in the full version of the paper.

In the notations hereunder, for any positive integers $\mathfrak{n}$, and $q \geq 2$, we define the "powers-of-2" matrix $\mathbf{H}_{\mathfrak{n}\times\mathfrak{n}\lceil\log q\rceil} = \mathbf{I}_{\mathfrak{n}} \otimes [1 \mid 2 \mid 4 \mid \ldots \mid 2^{\lceil\log q\rceil-1}] \in \mathbb{Z}_q^{\mathfrak{n}\times\mathfrak{n}\lceil\log q\rceil}$. Also, for each vector $\mathbf{v} \in \mathbb{Z}_q^{\mathfrak{n}}$, we define $\mathsf{bin}(\mathbf{v}) \in \{0,1\}^{\mathfrak{n}\lceil\log q\rceil}$ to be the vector obtained by replacing each entry of $\mathbf{v}$ by its binary expansion. Hence, we have $\mathbf{v} = \mathbf{H}_{\mathfrak{n}\times\mathfrak{n}\lceil\log q\rceil} \cdot \mathsf{bin}(\mathbf{v})$ for any $\mathbf{v} \in \mathbb{Z}_q^{\mathfrak{n}}$.

In our scheme, each group membership certificate is a signature generated by the group manager on the user's public key. Since the group manager only needs to sign known (rather than committed) messages, we can use a simplified version of the signature, where the chameleon hash function does not need to choose the discrete Gaussian vector $\mathbf{s}$ with a larger standard deviation than other vectors.

A key component of the scheme is the two-message joining protocol whereby the group manager admits new group members by signing their public key. The first message is sent by the new user $\mathcal{U}_i$ who samples a membership secret consisting of a short vector $\mathbf{z}_i \hookleftarrow D_{\mathbb{Z}^{4m},\sigma}$ (where $m = 2n\lceil\log q\rceil$), which is used to compute a syndrome $\mathbf{v}_i = \mathbf{F} \cdot \mathbf{z}_i \in \mathbb{Z}_q^{4n}$ for some public matrix $\mathbf{F} \in \mathbb{Z}_q^{4n\times 4m}$. This syndrome $\mathbf{v}_i \in \mathbb{Z}_q^{4n}$ must be signed by $\mathcal{U}_i$ using his long term secret key $\mathsf{usk}[i]$ (as in [55,13], we assume that each user has a long-term key $\mathsf{upk}[i]$ for a digital signature, which is registered in some PKI) and will uniquely identify $\mathcal{U}_i$. In order to generate a membership certificate for $\mathbf{v}_i \in \mathbb{Z}_q^{4n}$, the group manager GM signs its binary expansion $\mathsf{bin}(\mathbf{v}_i) \in \{0,1\}^{4n\lceil\log q\rceil}$ using the scheme of Section 3.

Equipped with his membership certificate $(\tau, \mathbf{d}, \mathbf{s}) \in \{0,1\}^{\ell} \times \mathbb{Z}^{2m} \times \mathbb{Z}^{2m}$,

the new group member $\mathcal{U}_i$ can sign a message using a Stern-like protocol for demonstrating his knowledge of a valid certificate for which he also knows the secret key associated with the certified public key $\mathbf{v}_i \in \mathbb{Z}_q^{4n}$. This boils down to providing evidence that the membership certificate is a valid signature on some binary message $\mathsf{bin}(\mathbf{v}_i) \in \{0,1\}^{4n\lceil \log q \rceil}$ for which he also knows a short $\mathbf{z}_i \in \mathbb{Z}^{4m}$ such that $\mathbf{v}_i = \mathbf{H}_{4n\times 2m} \cdot \mathsf{bin}(\mathbf{v}_i) = \mathbf{F} \cdot \mathbf{z}_i \in \mathbb{Z}_q^{4n}$.

Interestingly, the process does not require any proof of knowledge of the membership secret $\mathbf{z}_i$ during the joining phase, which is round-optimal. Analogously to the Kiayias-Yung technique [54] and constructions based on structure-preserving signatures [2], the joining protocol thus remains secure in environments where many users want to register at the same time in concurrent sessions.

4.1 Description of the Scheme

Setup$(1^\lambda, 1^{N_{\mathsf{gs}}})$**:** Given a security parameter $\lambda > 0$ and the maximal expected number of group members $N_{\mathsf{gs}} = 2^\ell \in \mathsf{poly}(\lambda)$, choose lattice parameter $n = \mathcal{O}(\lambda)$; prime modulus $q = \widetilde{\mathcal{O}}(\ell n^3)$; dimension $m = 2n\lceil \log q \rceil$; Gaussian parameter $\sigma = \Omega(\sqrt{n \log q} \log n)$; infinity norm bounds $\beta = \sigma\omega(\log m)$ and $B = \sqrt{n}\omega(\log n)$. Let χ be a B-bounded distribution. Choose a hash function $H : \{0,1\}^* \to \{1,2,3\}^t$ for some $t = \omega(\log n)$, which will be modeled as a random oracle in the security analysis. Then, do the following.

1. Generate a key pair for the signature of Section 3.1 for signing single-block messages. Namely, run $\mathsf{TrapGen}(1^n, 1^m, q)$ to get $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and a short basis $\mathbf{T_A}$ of $\Lambda_q^\perp(\mathbf{A})$, which allows computing short vectors in $\Lambda_q^\perp(\mathbf{A})$ with Gaussian parameter σ. Next, choose matrices $\mathbf{A}_0, \mathbf{A}_1, \ldots, \mathbf{A}_\ell, \mathbf{D} \hookleftarrow U(\mathbb{Z}_q^{n\times m})$, $\mathbf{D}_0, \mathbf{D}_1 \hookleftarrow U(\mathbb{Z}_q^{2n\times 2m})$ and a vector $\mathbf{u} \hookleftarrow U(\mathbb{Z}_q^n)$.
2. Choose an additional random matrix $\mathbf{F} \hookleftarrow U(\mathbb{Z}_q^{4n\times 4m})$ uniformly. Looking ahead, this matrix will be used to ensure security against framing attacks.
3. Generate a master key pair for the Gentry-Peikert-Vaikuntanathan IBE scheme in its multi-bit variant. This key pair consists of a statistically uniform matrix $\mathbf{B} \in \mathbb{Z}_q^{n\times m}$ and a short basis $\mathbf{T_B} \in \mathbb{Z}^{m\times m}$ of $\Lambda_q^\perp(\mathbf{B})$. This basis will allow us to compute GPV private keys with a Gaussian parameter $\sigma_{\mathrm{GPV}} \geq \|\widetilde{\mathbf{T}}_\mathbf{B}\| \cdot \sqrt{\log m}$.
4. Choose a one-time signature scheme $\Pi^{\mathrm{OTS}} = (\mathcal{G}, \mathcal{S}, \mathcal{V})$ and a hash function $H_0 : \{0,1\}^* \to \mathbb{Z}_q^{n\times 2m}$, that will be modeled as random oracles.

The group public key is defined as

$$\mathcal{Y} := \big(\mathbf{A},\ \{\mathbf{A}_j\}_{j=0}^\ell,\ \mathbf{B},\ \mathbf{D},\ \mathbf{D}_0,\ \mathbf{D}_1,\ \mathbf{F},\ \mathbf{u},\ \Pi^{\mathrm{OTS}},\ H,\ H_0\big).$$

The opening authority's private key is $\mathcal{S}_{\mathsf{OA}} := \mathbf{T_B}$ and the private key of the group manager consists of $\mathcal{S}_{\mathsf{GM}} := \mathbf{T_A}$. The algorithm outputs $\big(\mathcal{Y}, \mathcal{S}_{\mathsf{GM}}, \mathcal{S}_{\mathsf{OA}}\big)$.

Join$^{(\mathsf{GM},\mathcal{U}_i)}$**:** the group manager GM and the prospective user $\mathcal{U}_i$ run the following interactive protocol: $[\mathsf{J}_{\mathsf{user}}(\lambda, \mathcal{Y}), \mathsf{J}_{\mathsf{GM}}(\lambda, St, \mathcal{Y}, \mathcal{S}_{\mathsf{GM}})]$

1. $\mathcal{U}_i$ samples $\mathbf{z}_i \leftarrow D_{\mathbb{Z}^{4m},\sigma}$ and computes $\mathbf{v}_i = \mathbf{F} \cdot \mathbf{z}_i \in \mathbb{Z}_q^{4n}$. He sends the vector $\mathbf{v}_i \in \mathbb{Z}_q^{4n}$, whose binary representation is $\mathsf{bin}(\mathbf{v}_i) \in \{0,1\}^{2m}$, together with an ordinary digital signature $sig_i = \mathrm{Sign}_{\mathsf{usk}[i]}(\mathbf{v}_i)$ to GM.

2. $\mathsf{J_{GM}}$ verifies that $\mathbf{v}_i$ was not previously used by a registered user and that sig_i is a valid signature on $\mathbf{v}_i$ w.r.t. $\mathsf{upk}[i]$. It aborts if this is not the case. Otherwise, GM chooses a fresh identifier $\mathrm{id}_i \in \{0,1\}^\ell$ and uses $\mathcal{S}_{\mathsf{GM}} = \mathbf{T_A}$ to certify $\mathcal{U}_i$ as a new group member. To this end, GM defines

$$\mathbf{A}_{\mathrm{id}_i} = \Big[\, \mathbf{A} \;\Big|\; \mathbf{A}_0 + \textstyle\sum_{j=1}^{\ell} \mathrm{id}_i[j]\mathbf{A}_j \,\Big] \in \mathbb{Z}_q^{n\times 2m}. \tag{6}$$

Then, GM runs $\mathbf{T}'_{\mathrm{id}_i} \leftarrow \mathsf{ExtBasis}(\mathbf{A}_{\mathrm{id}_i}, \mathbf{T_A})$ to obtain a short delegated basis $\mathbf{T}'_{\mathrm{id}_i}$ of $\Lambda_q^{\perp}(\mathbf{A}_{\mathrm{id}_i}) \in \mathbb{Z}^{2m\times 2m}$. Finally, GM samples a short vector $\mathbf{s}_i \hookleftarrow D_{\mathbb{Z}^{2m},\sigma}$ and uses the obtained delegated basis $\mathbf{T}'_{\mathrm{id}_i}$ to compute a short vector $\mathbf{d}_i = [\mathbf{d}_{i,1}^T | \mathbf{d}_{i,2}^T]^T \in \mathbb{Z}^{2m}$ such that

$$\begin{aligned}\mathbf{A}_{\mathrm{id}_i} \cdot \mathbf{d}_i &= \Big[\, \mathbf{A} \;\Big|\; \mathbf{A}_0 + \textstyle\sum_{j=1}^{\ell} \mathrm{id}_i[j]\mathbf{A}_j \,\Big] \cdot \mathbf{d}_i \\ &= \mathbf{u} + \mathbf{D} \cdot \mathsf{bin}\big(\mathbf{D}_0 \cdot \mathsf{bin}(\mathbf{v}_i) + \mathbf{D}_1 \cdot \mathbf{s}_i\big) \bmod q. \end{aligned} \tag{7}$$

The triple $(\mathrm{id}_i, \mathbf{d}_i, \mathbf{s}_i)$ is sent to $\mathcal{U}_i$. Then, $\mathsf{J_{user}}$ verifies that the received $(\mathrm{id}_i, \mathbf{d}_i, \mathbf{s}_i)$ satisfies (7) and that $\|\mathbf{d}_i\|_\infty \leq \beta$, $\|\mathbf{s}_i\|_\infty \leq \beta$. If these conditions are not satisfied, $\mathsf{J_{user}}$ aborts. Otherwise, $\mathsf{J_{user}}$ defines the membership certificate as $\mathsf{cert}_i = (\mathrm{id}_i, \mathbf{d}_i, \mathbf{s}_i)$. The membership secret sec_i is defined to be $\mathsf{sec}_i = \mathbf{z}_i \in \mathbb{Z}^{4m}$. $\mathsf{J_{GM}}$ stores $\mathsf{transcript}_i = (\mathbf{v}_i, \mathsf{cert}_i, i, \mathsf{upk}[i], sig_i)$ in the database St_{trans} of joining transcripts.

$\mathsf{Sign}(\mathcal{Y}, \mathsf{cert}_i, \mathsf{sec}_i, M)$: To sign M using $\mathsf{cert}_i = (\mathrm{id}_i, \mathbf{d}_i, \mathbf{s}_i)$, where $\mathbf{d}_i \in \mathbb{Z}^{2m}$ and $\mathbf{s}_i \in \mathbb{Z}^{2m}$, as well as the membership secret $\mathsf{sec}_i = \mathbf{z}_i \in \mathbb{Z}^{4m}$, $\mathcal{U}_i$ generates a one-time signature key pair $(\mathsf{VK}, \mathsf{SK}) \leftarrow \mathcal{G}(n)$ and does the following.

1. Compute $\mathbf{G}_0 = H_0(\mathsf{VK}) \in \mathbb{Z}_q^{n\times 2m}$ and use it as an IBE public key to encrypt $\mathsf{bin}(\mathbf{v}_i) \in \{0,1\}^{2m}$, where $\mathbf{v}_i = \mathbf{F} \cdot \mathbf{z}_i \in \mathbb{Z}_q^{4n}$ is the syndrome of $\mathsf{sec}_i = \mathbf{z}_i \in \mathbb{Z}^{4m}$ for the matrix $\mathbf{F}$. Namely, compute $\mathbf{c}_{\mathbf{v}_i} \in \mathbb{Z}_q^m \times \mathbb{Z}_q^{2m}$ as

$$\mathbf{c}_{\mathbf{v}_i} = (\mathbf{c}_1, \mathbf{c}_2) = \big(\mathbf{B}^T \cdot \mathbf{e}_0 + \mathbf{x}_1,\ \mathbf{G}_0^T \cdot \mathbf{e}_0 + \mathbf{x}_2 + \mathsf{bin}(\mathbf{v}_i) \cdot \lfloor q/2 \rfloor\big) \tag{8}$$

for randomly chosen $\mathbf{e}_0 \hookleftarrow \chi^n$, $\mathbf{x}_1 \hookleftarrow \chi^m, \mathbf{x}_2 \hookleftarrow \chi^{2m}$. Notice that, as in the construction of [62], the columns of $\mathbf{G}_0$ can be interpreted as public keys for the multi-bit version of the dual Regev encryption scheme.

2. Run the protocol in Section 5.5 to prove the knowledge of $\mathrm{id}_i \in \{0,1\}^\ell$, vectors $\mathbf{s}_i \in \mathbb{Z}^{2m}, \mathbf{d}_{i,1}, \mathbf{d}_{i,2} \in \mathbb{Z}^m, \mathbf{z}_i \in \mathbb{Z}^{4m}$ with infinity norm bound β; $\mathbf{e}_0 \in \mathbb{Z}^n$, $\mathbf{x}_1 \in \mathbb{Z}^m, \mathbf{x}_2 \in \mathbb{Z}^{2m}$ with infinity norm bound B and $\mathsf{bin}(\mathbf{v}_i) \in \{0,1\}^{2m}, \mathbf{w}_i \in \{0,1\}^m$, that satisfy (8) as well as

$$\mathbf{A} \cdot \mathbf{d}_{i,1} + \mathbf{A}_0 \cdot \mathbf{d}_{i,2} + \sum_{j=1}^{\ell} (\mathrm{id}_i[j] \cdot \mathbf{d}_{i,2}) \cdot \mathbf{A}_j - \mathbf{D} \cdot \mathbf{w}_i = \mathbf{u} \in \mathbb{Z}_q^n \tag{9}$$

and

$$\begin{cases} \mathbf{H}_{2n\times m} \cdot \mathbf{w}_i = \mathbf{D}_0 \cdot \mathsf{bin}(\mathbf{v}_i) + \mathbf{D}_1 \cdot \mathbf{s}_i \in \mathbb{Z}_q^{2n} \\ \mathbf{F} \cdot \mathbf{z}_i = \mathbf{H}_{4n\times 2m} \cdot \mathsf{bin}(\mathbf{v}_i) \in \mathbb{Z}_q^{4n}. \end{cases} \tag{10}$$

The protocol is repeated $t = \omega(\log n)$ times in parallel to achieve negligible soundness error, and then made non-interactive using the Fiat-Shamir heuristic [40] as a triple $\pi_K = (\{\mathsf{Comm}_{K,j}\}_{j=1}^t, \mathsf{Chall}_K, \{\mathsf{Resp}_{K,j}\}_{j=1}^t)$, where $\mathsf{Chall}_K = H(M, \mathsf{VK}, \mathbf{c}_{\mathbf{v}_i}, \{\mathsf{Comm}_{K,j}\}_{j=1}^t) \in \{1,2,3\}^t$

3. Compute a one-time signature $sig = \mathcal{S}(\mathsf{SK}, (\mathbf{c}_{\mathbf{v}_i}, \pi_K))$.

Output the signature that consists of

$$\Sigma = \big(\mathsf{VK}, \mathbf{c}_{\mathbf{v}_i}, \pi_K, sig\big). \tag{11}$$

Verify$(\mathcal{Y}, M, \Sigma)$**:** Parse the signature Σ as in (11). Then, return 1 if and only if: (i) $\mathcal{V}(\mathsf{VK}, (\mathbf{c}_{\mathbf{v}_i}, \mathbf{c}_{\mathbf{s}_i}, \mathbf{c}_{\mathrm{id}}, \pi_K), sig) = 1$; (ii) The proof π_K properly verifies.

Open$(\mathcal{Y}, \mathcal{S}_{\mathsf{OA}}, M, \Sigma)$**:** Parse $\mathcal{S}_{\mathsf{OA}}$ as $\mathbf{T}_{\mathbf{B}} \in \mathbb{Z}^{m \times m}$ and Σ as in (11).

1. Compute $\mathbf{G}_0 = H_0(\mathsf{VK}) \in \mathbb{Z}_q^{n \times 2m}$. Then, using $\mathbf{T}_{\mathbf{B}}$ to compute a small-norm matrix $\mathbf{E}_{0,\mathsf{VK}} \in \mathbb{Z}^{m \times 2m}$ such that $\mathbf{B} \cdot \mathbf{E}_{0,\mathsf{VK}} = \mathbf{G}_0 \bmod q$.
2. Using $\mathbf{E}_{0,\mathsf{VK}}$, decrypt $\mathbf{c}_{\mathbf{v}_i}$ to obtain a string $\mathsf{bin}(\mathbf{v}) \in \{0,1\}^{2m}$ (i.e., by computing $\lfloor (\mathbf{c}_2 - \mathbf{E}_{0,\mathsf{VK}}^T \cdot \mathbf{c}_1)/(q/2) \rceil$).
3. Determine if the $\mathsf{bin}(\mathbf{v}) \in \{0,1\}^{2m}$ obtained at step 2 corresponds to a vector $\mathbf{v} = \mathbf{H}_{4n \times 2m} \cdot \mathsf{bin}(\mathbf{v}) \bmod q$ that appears in a record $\mathsf{transcript}_i = (\mathbf{v}, \mathsf{cert}_i, i, \mathsf{upk}[i], sig_i)$ of the database St_{trans} for some i. If so, output the corresponding i (and, optionally, $\mathsf{upk}[i]$). Otherwise, output $\perp$.

We remark that the scheme readily extends to provide a mechanism whereby the opening authority can efficiently prove that signatures were correctly opened at each opening operation. The difference between the dynamic group signature models suggested by Kiayias and Yung [55] and Bellare *et al.* [13] is that, in the latter, the opening authority (OA) must be able to convince a judge that the Open algorithm was run correctly. Here, such a mechanism can be realized using the techniques of public-key encryption with non-interactive opening [37]. Namely, since $\mathsf{bin}(\mathbf{v}_i)$ is encrypted using an IBE scheme for the identity VK, the OA can simply reveal the decryption matrix $\mathbf{E}_{0,\mathsf{VK}}$, that satisfies $\mathbf{B} \cdot \mathbf{E}_{0,\mathsf{VK}} = \mathbf{G}_0 \bmod q$ (which corresponds to the verification of a GPV signature) and allows the verifier to perform step 2 of the opening algorithm himself. The resulting construction is easily seen to satisfy the notion of opening soundness of Sakai *et al.* [71].

4.2 Efficiency and Correctness

Efficiency. The given dynamic group signature scheme can be implemented in polynomial time. The group public key has total bit-size $\mathcal{O}(\ell n m \log q) = \widetilde{\mathcal{O}}(\lambda^2) \cdot \log N_{\mathsf{gs}}$. The secret signing key of each user consists of a small constant number of low-norm vectors, and has bit-size $\widetilde{\mathcal{O}}(\lambda)$.

The size of each group signature is largely dominated by that of the non-interactive argument π_K, which is obtained from the Stern-like protocol of Section 5.5. Each round of the protocol has communication cost $\widetilde{\mathcal{O}}(m \cdot \log q) \cdot \log N_{\mathsf{gs}}$. Thus, the bit-size of π_K is $t \cdot \widetilde{\mathcal{O}}(m \cdot \log q) \cdot \log N_{\mathsf{gs}} = \widetilde{\mathcal{O}}(\lambda) \cdot \log N_{\mathsf{gs}}$. This is also the asymptotic bound on the size of the group signature.

Correctness. The correctness of algorithm $\mathsf{Verify}(\mathcal{Y}, M, \Sigma)$ follows from the facts that every certified group member is able to compute valid witness vectors satisfying equations (8), (9) and (10), and that the underlying argument system is perfectly complete. Moreover, the scheme parameters are chosen so that the GPV IBE [42] is correct, which implies that algorithm $\mathsf{Open}(\mathcal{Y}, \mathcal{S}_{\mathsf{OA}}, M, \Sigma)$ is also correct.

4.3 Security Analysis

Due to the fact that the number of public matrices $\{\mathbf{A}_j\}_{j=0}^{\ell}$ is only logarithmic in $N_{\mathsf{gs}} = 2^{\ell}$ instead of being linear in the security parameter λ, the proof of security against misidentification attacks (as defined in the full version of this paper and in [53]) cannot rely on the security of our signature scheme in a modular manner. The reason is that, at each run of the Join protocol, the group manager maintains a state and, instead of choosing the ℓ-bit identifier id uniformly in $\{0,1\}^{\ell}$, it chooses an identifier that has not been used yet. Since $\ell \ll \lambda$ (given that $N_{\mathsf{gs}} = 2^{\ell}$ is polynomial in λ), we thus have to prove security from scratch. However, the strategy of the reduction is exactly the same as in the security proof of the signature scheme.

The proofs of the following theorems are given in the full version of the paper.

Theorem 4. *The scheme is secure against misidentification attacks under the* $\mathsf{SIS}_{n,2m,q,\beta'}$ *assumption, for* $\beta' = \mathcal{O}(\ell\sigma^2 m^{3/2})$.

Theorem 5. *The scheme is secure against framing attacks under the* $\mathsf{SIS}_{4n,4m,q,\beta''}$ *assumption, where* $\beta'' = 4\sigma\sqrt{m}$.

Theorem 6. *In the random oracle model, the scheme provides CCA-anonymity if the* $\mathsf{LWE}_{n,q,\chi}$ *assumption holds and if* Π^{OTS} *is a strongly unforgeable one-time signature.*

5 Supporting Zero-Knowledge Argument Systems

This section provides a general framework that allows obtaining zero-knowledge arguments of knowledge (ZKAoK) for many relations appearing in lattice-based cryptography. Since lattice-based cryptosystems are built upon the hardness of the SIS and LWE problems, the relations among objects of the schemes are typically represented by modular linear equations. Thanks to the linearity property, we can often unify the given equations into one equation of the form:

$$\mathbf{P} \cdot \mathbf{x} = \mathbf{v} \bmod q, \tag{12}$$

where $(\mathbf{P}, \mathbf{v})$ are public and $\mathbf{x}$ is a secret vector (or matrix) that possesses some constraints to be proven in zero-knowledge, e.g., its smallness (like a SIS solution or an LWE noise) or a special arrangement of its entries. Starting from this high-level observation, we look for a tool that handles these constraints well.

Stern's protocol [72], originally proposed in the context of code-based cryptography, appears to be well-suited for our purpose. Stern's main idea is simple, yet elegant: To prove that a binary vector $\mathbf{x}$ has the fixed-Hamming-weight constraint, simply send the verifier a random permutation $\pi(\mathbf{x})$ which should guarantee that the constraint is satisfied while leaking no additional information about $\mathbf{x}$. Ling *et al.* [61] developed this idea to handle the smallness constraint, via a technique called Decomposition-Extension. This technique decomposes a vector with small infinity norm $B \geq 1$ into $\lfloor \log_2 B \rfloor + 1$ vectors with infinity norm 1, and then,

extends these vectors into elements of sets that are closed under permutations. Several subsequent works [57][62][60] employed the techniques of [72,61] in different contexts, but did not address the applicability and flexibility of the protocol in an abstract, generalized manner.

In Section 5.1, we abstract Stern's protocol to capture many relations that naturally appear in lattice-based cryptography. In particular, the argument systems used in our signature with efficient protocols (Section 3) and dynamic group signature (Section 4) can all be derived from this abstract protocol, which we will demonstrate in Sections 5.3, 5.4 and 5.5, respectively.

We note that several works [51,73,15] addressed the problem of proving multiplicative and additive relations among committed linear objects (matrices and vectors over $\mathbb{Z}_q$) in lattice-based cryptography. These results, however, do not yield a simple solution for the relations involved in our schemes. If we were to plug proof systems like [51,73,15] in our relations, we would need to commit to all objects using perfectly binding commitments (which would require very long commitment keys) and express the relations in terms of many multiplications and additions gates before running many instances of the proof systems depending on the circuit. Instead of considering general circuits, our framework aims at a more direct (but still fairly general) solution for a large class of relations that naturally appear in SIS and LWE-based cryptography.

5.1 Abstracting Stern's Protocol

Let $D, L, q \geq 2$ be positive integers let VALID be a subset of $\{-1, 0, 1\}^L$. Suppose that $\mathcal{S}$ is a finite set such that one can associate every $\pi \in \mathcal{S}$ with a permutation T_π of L elements, satisfying the following conditions:

$$\begin{cases} \mathbf{x} \in \mathsf{VALID} \iff T_\pi(\mathbf{x}) \in \mathsf{VALID}, \\ \text{If } \mathbf{x} \in \mathsf{VALID} \text{ and } \pi \text{ is uniform in } \mathcal{S}, \text{ then } T_\pi(\mathbf{x}) \text{ is uniform in } \mathsf{VALID}. \end{cases} \tag{13}$$

We aim to construct a statistical ZKAoK for the following abstract relation:

$$\mathrm{R}_{\mathrm{abstract}} = \left\{ (\mathbf{P}, \mathbf{v}), \mathbf{x} \in \mathbb{Z}_q^{D \times L} \times \mathbb{Z}_q^D \times \mathsf{VALID} : \mathbf{P} \cdot \mathbf{x} = \mathbf{v} \bmod q. \right\}$$

Note that, Stern's original protocol corresponds to the special case when $\mathsf{VALID} = \{\mathbf{x} \in \{0,1\}^L : \mathsf{wt}(\mathbf{x}) = k\}$ (where $\mathsf{wt}(\cdot)$ denotes the Hamming weight and $k < L$ is a given integer), $\mathcal{S} = \mathcal{S}_L$ - hereunder the set of all permutations of L elements, and $T_\pi(\mathbf{x}) = \pi(\mathbf{x})$.

The conditions in (13) play a crucial role in proving in ZK that $\mathbf{x} \in \mathsf{VALID}$: To do so, the prover samples $\pi \hookleftarrow U(\mathcal{S})$ and let the verifier check that $T_\pi(\mathbf{x}) \in \mathsf{VALID}$, while the latter cannot learn any additional information about $\mathbf{x}$ thanks to the randomness of π. Furthermore, to prove in ZK that the linear equation holds, the prover samples a masking vector $\mathbf{r} \hookleftarrow U(\mathbb{Z}_q^L)$, sends $\mathbf{y} = \mathbf{x} + \mathbf{r} \bmod q$, and convinces the verifier instead that $\mathbf{P} \cdot \mathbf{y} = \mathbf{P} \cdot \mathbf{r} + \mathbf{v} \bmod q$.

The interactive protocol between the prover and the verifier with common input $(\mathbf{P}, \mathbf{v})$ and prover's secret input $\mathbf{x}$ is described in Figure 1. The protocol

1. **Commitment:** Prover samples $\mathbf{r} \hookleftarrow U(\mathbb{Z}_q^L)$, $\pi \hookleftarrow U(\mathcal{S})$ and randomness ρ_1, ρ_2, ρ_3 for COM. Then he sends $\mathrm{CMT} = \big(C_1, C_2, C_3\big)$ to the verifier, where

$$C_1 = \mathsf{COM}(\pi, \mathbf{P} \cdot \mathbf{r}; \rho_1), \;\; C_2 = \mathsf{COM}(T_\pi(\mathbf{r}); \rho_2), \;\; C_3 = \mathsf{COM}(T_\pi(\mathbf{x} + \mathbf{r}); \rho_3).$$

2. **Challenge:** The verifier sends a challenge $Ch \hookleftarrow U(\{1, 2, 3\})$ to the prover.
3. **Response:** Depending on Ch, the prover sends RSP computed as follows:
   - $Ch = 1$: Let $\mathbf{t}_x = T_\pi(\mathbf{x})$, $\mathbf{t}_r = T_\pi(\mathbf{r})$, and $\mathrm{RSP} = (\mathbf{t}_x, \mathbf{t}_r, \rho_2, \rho_3)$.
   - $Ch = 2$: Let $\pi_2 = \pi$, $\mathbf{y} = \mathbf{x} + \mathbf{r}$, and $\mathrm{RSP} = (\pi_2, \mathbf{y}, \rho_1, \rho_3)$.
   - $Ch = 3$: Let $\pi_3 = \pi$, $\mathbf{r}_3 = \mathbf{r}$, and $\mathrm{RSP} = (\pi_3, \mathbf{r}_3, \rho_1, \rho_2)$.

Verification: Receiving RSP, the verifier proceeds as follows:

- $Ch = 1$: Check that $\mathbf{t}_x \in \mathsf{VALID}$ and $C_2 = \mathsf{COM}(\mathbf{t}_r; \rho_2)$, $C_3 = \mathsf{COM}(\mathbf{t}_x + \mathbf{t}_r; \rho_3)$.
- $Ch = 2$: Check that $C_1 = \mathsf{COM}(\pi_2, \mathbf{P} \cdot \mathbf{y} - \mathbf{v}; \rho_1)$, $C_3 = \mathsf{COM}(T_{\pi_2}(\mathbf{y}); \rho_3)$.
- $Ch = 3$: Check that $C_1 = \mathsf{COM}(\pi_3, \mathbf{P} \cdot \mathbf{r}_3; \rho_1)$, $\;C_2 = \mathsf{COM}(T_{\pi_3}(\mathbf{r}_3); \rho_2)$.

In each case, the verifier outputs 1 if and only if all the conditions hold.

Fig. 1: A ZKAoK for the relation $\mathrm{R}_{\mathrm{abstract}}$.

employs a statistically hiding and computationally binding string commitment scheme COM (e.g., the SIS-based one from [52]).

The properties of the given protocol are summarized in the following lemma.

Lemma 8. *The protocol in Figure 1 is a statistical* ZKAoK *for the relation* $\mathrm{R}_{\mathrm{abstract}}$ *with perfect completeness, soundness error* $2/3$*, and communication cost* $\widetilde{\mathcal{O}}(L \log q)$*. In particular:*

- *There exists an efficient simulator that, on input* $(\mathbf{P}, \mathbf{v})$*, outputs an accepted transcript which is statistically close to that produced by the real prover.*
- *There exists an efficient knowledge extractor that, on input a commitment* CMT *and* 3 *valid responses* $(\mathrm{RSP}_1, \mathrm{RSP}_2, \mathrm{RSP}_3)$ *to all* 3 *possible values of the challenge* Ch*, outputs* $\mathbf{x}' \in \mathsf{VALID}$ *such that* $\mathbf{P} \cdot \mathbf{x}' = \mathbf{v} \bmod q$*.*

The proof of Lemma 8 employs standard simulation and extraction techniques for Stern-like protocols [52][61][62]. It is detailed in the full version of the paper.

5.2 Supporting Notations and Techniques

Below we will describe the notations and techniques, adapted from recent works on Stern-like protocols [61][57][39][60], that we will employ in the next subsections to handle 3 different constraints of the witness vectors.

Let m be an arbitrary dimension, and B be an arbitrary infinity norm bound.

Case 1: $\mathbf{w} \in \{0, 1\}^m$. We denote by B_m^2 the set of all vectors in $\{0, 1\}^{2m}$ having exactly m coordinates equal to 1. We also let $\mathsf{Ext}_{2m}(\mathbf{w})$ be the algorithm that outputs a vector $\hat{\mathbf{w}} \in \mathsf{B}_m^2$ by appending m suitable coordinates to $\mathbf{w} \in \{0, 1\}^m$. Note that, for any permutation $\rho \in \mathcal{S}_{2m}$, we have $\hat{\mathbf{w}} \in \mathsf{B}_m^2 \Leftrightarrow \rho(\hat{\mathbf{w}}) \in \mathsf{B}_m^2$.

Case 2: $\mathbf{w} \in [-B, B]^m$. We define $\delta_B := \lfloor \log_2 B \rfloor + 1$ and denote by $\mathsf{B}^3_{m\delta_B}$ the set of vectors in $\{-1, 0, 1\}^{3m\delta_B}$ with exactly $m\delta_B$ coordinates equal to j, for every $j \in \{-1, 0, 1\}$. The Decomposition-Extension technique from [61] consists in transforming $\mathbf{w} \in [-B, B]^m$ to a vector $\mathsf{DecExt}_{m,B}(\mathbf{w}) \in \mathsf{B}^3_{m\delta_B}$, as follows.

Define the sequence $B_1, \ldots, B_{\delta_B}$, where $B_j = \left\lfloor \frac{B+2^{j-1}}{2^j} \right\rfloor$ for all $j \in [1, \delta_B]$. As noted in [61], it satisfies $\sum_{j=1}^{\delta_B} B_j = B$, and for any $w \in [-B, B]$, one can efficiently compute $w^{(1)}, \ldots, w^{(\delta_B)} \in \{-1, 0, 1\}$ such that $\sum_{j=1}^{\delta_B} B_j \cdot w^{(j)} = w$. Next, define the matrix

$$\mathbf{K}_{m,B} = \mathbf{I}_m \otimes [B_1 \,|\, \ldots \,|\, B_{\delta_B}] = \begin{bmatrix} B_1 \ldots B_{\delta_B} & & \\ & \ddots & \\ & & B_1 \ldots B_{\delta_B} \end{bmatrix} \in \mathbb{Z}^{m \times m\delta_B},$$

and its extension $\widehat{\mathbf{K}}_{m,B} = \left[\mathbf{K}_{m,B} \middle| \mathbf{0}^{m \times 2m\delta_B}\right] \in \mathbb{Z}^{m \times 3m\delta_B}$.

If we let $\mathbf{w} = (w_1, \ldots, w_m)^T$, then we can compute

$$\mathbf{w}' = \left(w_1^{(1)}, \ldots, w_1^{(\delta_B)}, \ldots, w_m^{(1)}, \ldots, w_m^{(\delta_B)}\right)^T \in \{-1, 0, 1\}^{m\delta_B}$$

satisfying $\mathbf{K}_{m,B} \cdot \mathbf{w}' = \mathbf{w}$. By appending $2m\delta_B$ suitable coordinates to $\mathbf{w}'$, we can obtain $\hat{\mathbf{w}} \in \mathsf{B}^3_{m\delta_B}$ satisfying $\widehat{\mathbf{K}}_{m,B} \cdot \hat{\mathbf{w}} = \mathbf{w}$.

Note that for any $\phi \in \mathcal{S}_{3m\delta_B}$, we have $\hat{\mathbf{w}} \in \mathsf{B}^3_{m\delta_B} \Leftrightarrow \phi(\hat{\mathbf{w}}) \in \mathsf{B}^3_{m\delta_B}$.

Case 3: $\mathbf{w} \in \{0, 1\}^{2m}$ is the correct encoding of some $\mathbf{t} \in \{0, 1\}^m$.

Recall that the encoding function from Section 3.3, hereunder denoted by Encode_m if the input is a binary vector of length m, extends $\mathbf{t} = (t_1, \ldots, t_m)^T$ to $\mathsf{Encode}_m(\mathbf{t}) = (\bar{t}_1, t_1, \ldots, \bar{t}_m, t_m)$. We define $\mathsf{CorEnc}(m) = \{\mathbf{w} = \mathsf{Encode}_m(\mathbf{t}) : \mathbf{t} \in \{0, 1\}^m\}$ - the set of all correct encodings of m-bit vectors. To handle the constraint $\mathbf{w} \in \mathsf{CorEnc}(m)$, we adapt the permuting technique from [57][39][60].

For $\mathbf{b} = (b_1, \ldots, b_m)^T \in \{0, 1\}^m$, we let $E_{\mathbf{b}}$ be the permutation transforming vector $\mathbf{w} = (w_1^0, w_1^1, \ldots, w_m^0, w_m^1) \in \mathbb{Z}^{2m}$ to $E_{\mathbf{b}}(\mathbf{w}) = (w_1^{b_1}, w_1^{\bar{b}_1}, \ldots, w_m^{b_m}, w_m^{\bar{b}_m})$. Note that, $E_{\mathbf{b}}$ transforms $\mathbf{w} = \mathsf{Encode}_m(\mathbf{t})$ to $E_{\mathbf{b}}(\mathbf{w}) = \mathsf{Encode}_m(\mathbf{t} \oplus \mathbf{b})$, where $\oplus$ denotes the bit-wise addition modulo 2. Thus, for any $\mathbf{b} \in \{0, 1\}^m$, we have

$$\mathbf{w} \in \mathsf{CorEnc}(m) \Leftrightarrow E_{\mathbf{b}}(\mathbf{w}) \in \mathsf{CorEnc}(m).$$

5.3 Proving the Consistency of Commitments

The argument system used in our protocol for signing a committed value in Section 3.3 can be summarized as follows.

Common Input: Matrices $\{\mathbf{D}_k \in \mathbb{Z}_q^{2n \times 2m}\}_{k=0}^N$; $\mathbf{B} \in \mathbb{Z}_q^{n \times m}$; $\mathbf{G}_1 \in \mathbb{Z}_q^{n \times 2m}$;
vectors $\mathbf{c}_{\mathfrak{m}} \in \mathbb{Z}_q^{2n}$; $\{\mathbf{c}_{k,1} \in \mathbb{Z}_q^m\}_{k=1}^N$; $\{\mathbf{c}_{k,2} \in \mathbb{Z}_q^{2m}\}_{k=1}^N$; $\mathbf{c}_{\mathbf{s}',1} \in \mathbb{Z}_q^m$; $\mathbf{c}_{\mathbf{s}',2} \in \mathbb{Z}_q^{2m}$.

Prover's Input: $\mathfrak{m} = (\mathfrak{m}_1^T \| \ldots \| \mathfrak{m}_N^T)^T \in \mathsf{CorEnc}(mN)$;
$\{\mathbf{s}_k \in [-B, B]^n,\ \mathbf{e}_{k,1} \in [-B, B]^m;\ \mathbf{e}_{k,2} \in [-B, B]^{2m}\}_{k=1}^N$; $\mathbf{s}_0 \in [-B, B]^n$;
$\mathbf{e}_{0,1} \in [-B, B]^m$; $\mathbf{e}_{0,2} \in [-B, B]^{2m}$; $\mathbf{s}' \in [-(p-1), (p-1)]^{2m}$

Prover's Goal: Convince the verifier in ZK that:

$$\begin{cases} \mathbf{c}_{\mathfrak{m}} = \mathbf{D}_0 \cdot \mathbf{s}' + \sum_{k=1}^{N} \mathbf{D}_k \cdot \mathfrak{m}_k \bmod q; \\ \mathbf{c}_{\mathbf{s}',1} = \mathbf{B}^T \cdot \mathbf{s}_0 + \mathbf{e}_{0,1} \bmod q; \;\; \mathbf{c}_{\mathbf{s}',2} = \mathbf{G}_1^T \cdot \mathbf{s}_0 + \mathbf{e}_{0,2} + \lfloor q/p \rfloor \cdot \mathbf{s}' \bmod q; \\ \forall k \in [N] : \mathbf{c}_{k,1} = \mathbf{B}^T \cdot \mathbf{s}_k + \mathbf{e}_{k,1}; \;\; \mathbf{c}_{k,2} = \mathbf{G}_1^T \cdot \mathbf{s}_k + \mathbf{e}_{k,2} + \lfloor q/2 \rfloor \cdot \mathfrak{m}_k. \end{cases} \quad (14)$$

We will show that the above argument system can be obtained from the one in Section 5.1. We proceed in two steps.

Step 1: *Transforming the equations in (14) into a unified one of the form* $\mathbf{P} \cdot \mathbf{x} = \mathbf{v} \bmod q$*, where* $\|\mathbf{x}\|_\infty = 1$ *and* $\mathbf{x} \in \mathsf{VALID}$ *- a "specially-designed" set.*

To do so, we first form the following vectors and matrices:

$$\begin{cases} \mathbf{x}_1 = \left(\mathbf{s}_0^T \| \mathbf{e}_{0,1}^T \| \mathbf{e}_{0,2}^T \| \mathbf{s}_1^T \| \mathbf{e}_{1,1}^T \| \mathbf{e}_{1,2}^T \| \ldots \| \mathbf{s}_N^T \| \mathbf{e}_{N,1}^T \| \mathbf{e}_{N,2}^T\right)^T \in [-B, B]^{(n+3m)(N+1)}; \\ \mathbf{v} = \left(\mathbf{c}_{\mathfrak{m}}^T \| \mathbf{c}_{\mathbf{s}',1}^T \| \mathbf{c}_{\mathbf{s}',2}^T \| \mathbf{c}_{1,1}^T \| \mathbf{c}_{1,2}^T \| \ldots \| \mathbf{c}_{N,1}^T \| \mathbf{c}_{N,2}^T\right)^T \in \mathbb{Z}_q^{2n+3m(N+1)}; \\ \mathbf{P}_1 = \left(\begin{array}{c|c} \mathbf{B}^T & \\ \hline \mathbf{G}_1^T & \end{array} \middle| \mathbf{I}_{3m}\right); \quad \mathbf{Q}_2 = \left(\begin{array}{c} \mathbf{0} \\ \hline \lfloor \frac{q}{2} \rfloor \mathbf{I}_{2m} \end{array}\right); \quad \mathbf{Q}_p = \left(\begin{array}{c} \mathbf{0} \\ \hline \lfloor \frac{q}{p} \rfloor \mathbf{I}_{2m} \end{array}\right) \\ \mathbf{M}_1 = \left(\begin{array}{c} \mathbf{0} \\ \hline \mathbf{P}_1 \\ \quad \mathbf{P}_1 \\ \quad\quad \ddots \\ \quad\quad\quad \mathbf{P}_1 \end{array}\right); \quad \mathbf{M}_2 = \left(\begin{array}{c} \mathbf{D}_1 \vert \ldots \vert \mathbf{D}_N \\ \hline \mathbf{0} \\ \hline \mathbf{Q}_2 \\ \quad \ddots \\ \quad\quad \mathbf{Q}_2 \end{array}\right); \quad \mathbf{M}_3 = \left(\begin{array}{c} \mathbf{D}_0 \\ \hline \mathbf{Q}_p \\ \hline \\ \mathbf{0} \\ \\ \end{array}\right). \end{cases}$$

We then observe that (14) can be rewritten as:

$$\mathbf{M}_1 \cdot \mathbf{x}_1 + \mathbf{M}_2 \cdot \mathfrak{m} + \mathbf{M}_3 \cdot \mathbf{s}' = \mathbf{v} \in \mathbb{Z}_q^D, \quad (15)$$

where $D = 2n + 3m(N+1)$. Now we employ the techniques from Section 5.2 to convert (15) into the form $\mathbf{P} \cdot \mathbf{x} = \mathbf{v} \bmod q$. Specifically, if we let:

$$\begin{cases} \mathsf{DecExt}_{(n+3m)(N+1),B}(\mathbf{x}_1) \to \hat{\mathbf{x}}_1 \in \mathsf{B}^3_{(n+3m)(N+1)\delta_B}; \\ \mathbf{M}_1' = \mathbf{M}_1 \cdot \widehat{\mathbf{K}}_{(n+3m)(N+1),B} \in \mathbb{Z}_q^{D \times 3(n+3m)(N+1)\delta_B}; \\ \mathsf{DecExt}_{2m,p-1}(\mathbf{s}') \to \hat{\mathbf{s}} \in \mathsf{B}^3_{2m\delta_{p-1}}; \;\; \mathbf{M}_3' = \mathbf{M}_3 \cdot \widehat{\mathbf{K}}_{2m,p-1} \in \mathbb{Z}_q^{D \times 6m\delta_{p-1}}, \end{cases}$$

$L = 3(n+3m)(N+1)\delta_B + 2mN + 6m\delta_{p-1}$, and $\mathbf{P} = \left[\mathbf{M}_1' | \mathbf{M}_2 | \mathbf{M}_3'\right] \in \mathbb{Z}_q^{D \times L}$, and $\mathbf{x} = \left(\hat{\mathbf{x}}_1^T \| \mathfrak{m}^T \| \hat{\mathbf{s}}^T\right)^T$, then we will obtain the desired equation:

$$\mathbf{P} \cdot \mathbf{x} = \mathbf{v} \bmod q.$$

Having performed the above unification, we now define VALID as the set of all vectors $\mathbf{t} \in \{-1, 0, 1\}^L$ of the form $\mathbf{t} = \left(\mathbf{t}_1^T \| \mathbf{t}_2^T \| \mathbf{t}_3^T\right)^T$, where $\mathbf{t}_1 \in \mathsf{B}^3_{(n+3m)(N+1)\delta_B}$, $\mathbf{t}_2 \in \mathsf{CorEnc}(mN)$, and $\mathbf{t}_3 \in \mathsf{B}^3_{2m\delta_{p-1}}$. Note that $\mathbf{x} \in \mathsf{VALID}$.

Step 2: *Specifying the set* $\mathcal{S}$ *and permutations of* L *elements* $\{T_\pi : \pi \in \mathcal{S}\}$ *for which the conditions in (13) hold.*

– Define $\mathcal{S} := \mathcal{S}_{3(n+3m)(N+1)\delta_B} \times \{0,1\}^{mN} \times \mathcal{S}_{6m\delta_{p-1}}$.

– For $\pi = (\pi_1, \mathbf{b}, \pi_3) \in \mathcal{S}$, and for vector $\mathbf{w} = \left(\mathbf{w}_1^T \| \mathbf{w}_2^T \| \mathbf{w}_3^T\right)^T \in \mathbb{Z}_q^L$, where $\mathbf{w}_1 \in \mathbb{Z}_q^{3(n+3m)(N+1)\delta_B}$, $\mathbf{w}_2 \in \mathbb{Z}_q^{2mN}$, $\mathbf{w}_3 \in \mathbb{Z}_q^{6m\delta_{p-1}}$, we define:

$$T_\pi = \left(\pi_1(\mathbf{w}_1)^T \| E_{\mathbf{b}}(\mathbf{w}_2)^T \| \pi_3(\mathbf{w}_3)^T\right)^T.$$

By inspection, it can be seen that the properties in (13) are satisfied, as desired. As a result, we can obtain the required argument system by running the protocol in Section 5.1 with common input $(\mathbf{P}, \mathbf{v})$ and prover's input $\mathbf{x}$.

5.4 Proving the Possession of a Signature on a Committed Value

We now describe how to derive the protocol for proving the possession of a signature on a committed value, that is used in Section 3.3.

Common Input: Matrices $\mathbf{A}, \{\mathbf{A}_j\}_{j=0}^{\ell}, \mathbf{D} \in \mathbb{Z}_q^{n\times m}$; $\{\mathbf{D}_k \in \mathbb{Z}_q^{2n\times 2m}\}_{k=0}^{N}$; $\mathbf{B} \in \mathbb{Z}_q^{n\times m}$; $\mathbf{G}_1 \in \mathbb{Z}_q^{n\times 2m}$; $\mathbf{G}_0 \in \mathbb{Z}_q^{n\times \ell}$; vectors $\{\mathbf{c}_{k,1}\}_{k=1}^{N}, \mathbf{c}_{\tau,1}, \mathbf{c}_{\mathbf{v},1}, \mathbf{c}_{s,1} \in \mathbb{Z}_q^m$; $\{\mathbf{c}_{k,2}\}_{k=1}^{N}, \mathbf{c}_{\mathbf{v},2}, \mathbf{c}_{s,2} \in \mathbb{Z}_q^{2m}$; $\mathbf{c}_{\tau,2} \in \mathbb{Z}_q^{\ell}$; $\mathbf{u} \in \mathbb{Z}_q^n$.

Prover's Input: $\mathbf{v} = \begin{pmatrix}\mathbf{v}_1\\ \mathbf{v}_2\end{pmatrix}$, where $\mathbf{v}_1, \mathbf{v}_2 \in [-\beta, \beta]^m$ and $\beta = \sigma \cdot \omega(\log m)$ - the infinity norm bound of signatures; $\tau \in \{0,1\}^{\ell}$; $\mathbf{s} \in [-(p-1), (p-1)]^{2m}$; $\mathfrak{m} = (\mathfrak{m}_1^T \| \ldots \| \mathfrak{m}_N^T)^T \in \mathsf{CorEnc}(mN)$; $\{\mathbf{s}_k\}_{k=1}^{N}$, $\mathbf{s}_{\mathbf{v}}$, $\mathbf{s}_0$, $\mathbf{s}_\tau \in [-B, B]^n$; $\{\mathbf{e}_{k,1}\}_{k=1}^{N}$, $\mathbf{e}_{\mathbf{v},1}$, $\mathbf{e}_{0,1}$, $\mathbf{e}_{\tau,1} \in [-B, B]^m$; $\{\mathbf{e}_{k,2}\}_{k=1}^{N}, \mathbf{e}_{0,2}, \mathbf{e}_{\mathbf{v},2} \in [-B, B]^{2m}$; $\mathbf{e}_{\tau,2} \in [-B, B]^{\ell}$.

Prover's Goal: Convince the verifier in ZK that:

$$\mathbf{A}\cdot\mathbf{v}_1 + \mathbf{A}_0\cdot\mathbf{v}_2 + \sum_{i=1}^{\ell}\mathbf{A}_i\cdot\tau[i]\mathbf{v}_2 - \mathbf{D}\cdot\mathsf{bin}(\mathbf{D}_0\cdot\mathbf{s} + \sum_{k=1}^{N}\mathbf{D}_i\cdot\mathfrak{m}_k) = \mathbf{u} \bmod q, \quad (16)$$

and that (modulo q)

$$\begin{cases} \forall k \in [N] : \mathbf{c}_{k,1} = \mathbf{B}^T\cdot\mathbf{s}_k + \mathbf{e}_{k,1};\ \ \mathbf{c}_{k,2} = \mathbf{G}_1^T\cdot\mathbf{s}_k + \mathbf{e}_{k,2} + \lfloor q/2\rfloor\cdot\mathfrak{m}_k; \\ \mathbf{c}_{\mathbf{v},1} = \mathbf{B}^T\cdot\mathbf{s}_{\mathbf{v}} + \mathbf{e}_{\mathbf{v},1}; \\ \mathbf{c}_{\mathbf{v},2} = \mathbf{G}_1^T\cdot\mathbf{s}_{\mathbf{v}} + \mathbf{e}_{\mathbf{v},2} + \lfloor\frac{q}{p}\rfloor\cdot\mathbf{v} = \mathbf{G}_1^T\cdot\mathbf{s}_{\mathbf{v}} + \mathbf{e}_{\mathbf{v},2} + \begin{pmatrix}\lfloor\frac{q}{p}\rfloor\mathbf{I}_m\\ \mathbf{0}\end{pmatrix}\cdot\mathbf{v}_1 + \begin{pmatrix}\mathbf{0}\\ \lfloor\frac{q}{p}\rfloor\mathbf{I}_m\end{pmatrix}\cdot\mathbf{v}_2; \\ \mathbf{c}_{\mathbf{s},1} = \mathbf{B}^T\cdot\mathbf{s}_0 + \mathbf{e}_{0,1};\ \ \mathbf{c}_{\mathbf{s},2} = \mathbf{G}_1^T\cdot\mathbf{s}_0 + \mathbf{e}_{0,2} + \lfloor q/p\rfloor\cdot\mathbf{s}; \\ \mathbf{c}_{\tau,1} = \mathbf{B}^T\cdot\mathbf{s}_\tau + \mathbf{e}_{\tau,1};\ \mathbf{c}_{\tau,2} = \mathbf{G}_0^T\cdot\mathbf{s}_\tau + \mathbf{e}_{\tau,2} + \lfloor q/2\rfloor\cdot\tau. \end{cases} \quad (17)$$

We proceed in two steps.

Step 1: *Transforming the equations in (16) and (17) into a unified one of the form* $\mathbf{P}\cdot\mathbf{x} = \mathbf{c} \bmod q$*, where* $\|\mathbf{x}\|_\infty = 1$ *and* $\mathbf{x} \in \mathsf{VALID}$ *- a "specially-designed" set.*

Note that, if we let $\mathbf{y} = \mathsf{bin}(\mathbf{D}_0 \cdot \mathbf{s} + \sum_{k=1}^{N} \mathbf{D}_i \cdot \mathfrak{m}_k) \in \{0,1\}^m$, then we have $\mathbf{H}_{2n\times m} \cdot \mathbf{y} = \mathbf{D}_0 \cdot \mathbf{s} + \sum_{k=1}^{N} \mathbf{D}_i \cdot \mathfrak{m}_k \bmod q$, and (16) can be equivalently written as:

$$\begin{pmatrix}\mathbf{A}\\ \mathbf{0}\end{pmatrix}\cdot\mathbf{v}_1 + \begin{pmatrix}\mathbf{A}_0\\ \mathbf{0}\end{pmatrix}\cdot\mathbf{v}_2 + \sum_{i=1}^{\ell}\begin{pmatrix}\mathbf{A}_i\\ \mathbf{0}\end{pmatrix}\cdot\tau[i]\mathbf{v}_2 + \begin{pmatrix}\mathbf{0}\\ \mathbf{D}_0\end{pmatrix}\cdot\mathbf{s} + \begin{pmatrix}-\mathbf{D}\\ -\mathbf{H}_{2n\times m}\end{pmatrix}\cdot\mathbf{y} + \begin{pmatrix}\mathbf{0}\\ \mathbf{D}_1|\ldots|\mathbf{D}_N\end{pmatrix}\cdot\mathfrak{m} = \begin{pmatrix}\mathbf{u}\\ \mathbf{0}^{2n}\end{pmatrix} \mod q.$$

Next, we use linear algebra to combine this equation and (17) into (modulo q):

$$\mathbf{F}\cdot\mathbf{v}_1 + \mathbf{F}_0\cdot\mathbf{v}_2 + \sum_{i=1}^{\ell}\mathbf{F}_i\cdot\tau[i]\mathbf{v}_2 + \mathbf{M}_1\cdot\tau + \mathbf{M}_2\cdot\mathbf{y} + \mathbf{M}_3\cdot\mathfrak{m} + \mathbf{M}_4\cdot\mathbf{s} + \mathbf{M}_5\cdot\mathbf{e} = \mathbf{c}, \quad (18)$$

where, for dimensions $D = \ell + 3n + 7m + 3mN$ and $L_0 = D + nN$,

- Matrices $\mathbf{F}, \mathbf{F}_0, \mathbf{F}_1, \ldots, \mathbf{F}_\ell \in \mathbb{Z}_q^{D\times m}$, $\mathbf{M}_1 \in \mathbb{Z}_q^{D\times \ell}$, $\mathbf{M}_2 \in \mathbb{Z}_q^{D\times m}$, $\mathbf{M}_3 \in \mathbb{Z}_q^{D\times 2mN}$, $\mathbf{M}_4 \in \mathbb{Z}_q^{D\times 2m}$, $\mathbf{M}_5 \in \mathbb{Z}_q^{D\times L_0}$ and vector $\mathbf{c} \in \mathbb{Z}_q^D$ are built from the public input.
- Vector $\mathbf{e} = \big(\mathbf{s}_1^T \| \ldots \| \mathbf{s}_N^T \| \mathbf{s}_{\mathbf{v}}^T \| \mathbf{s}_0^T \| \mathbf{s}_\tau^T \| \mathbf{e}_{1,1}^T \| \ldots \| \mathbf{e}_{N,1}^T \| \mathbf{e}_{\mathbf{v},1}^T \| \mathbf{e}_{0,1}^T \| \mathbf{e}_{\tau,1}^T \| \| \mathbf{e}_{1,2}^T \| \ldots \| \mathbf{e}_{N,2}^T \| \mathbf{e}_{0,2}^T \| \mathbf{e}_{\mathbf{v},2}^T \| \mathbf{e}_{\tau,2}^T\big)^T \in [-B, B]^{L_0}$.

Now we further transform (18) using the techniques from Section 5.2. Specifically, we form the following:

$$\begin{cases}
\mathsf{DecExt}_{m,\beta}(\mathbf{v}_1) \to \hat{\mathbf{v}}_1 \in \mathsf{B}_{m\delta_\beta}^3;\ \ \mathsf{DecExt}_{m,\beta}(\mathbf{v}_2) \to \hat{\mathbf{v}}_2 \in \mathsf{B}_{m\delta_\beta}^3;\\
\mathbf{F}' = \big[\mathbf{F}\cdot\widehat{\mathbf{K}}_{m,\beta} | \mathbf{F}_0\cdot\widehat{\mathbf{K}}_{m,\beta} | \mathbf{F}_1\cdot\widehat{\mathbf{K}}_{m,\beta} | \ldots | \mathbf{F}_\ell\cdot\widehat{\mathbf{K}}_{m,\beta} | \mathbf{0}^{D\times 3m\delta_\beta\ell}\big] \in \mathbb{Z}_q^{D\times 3m\delta_\beta(2\ell+2)};\\
\mathsf{Ext}_{2\ell}(\tau) \to \hat{\tau} = (\tau[1], \ldots, \tau[\ell], \ldots, \tau[2\ell])^T \in \mathsf{B}_\ell^2;\ \mathbf{M}_1' = [\mathbf{M}_1 | \mathbf{0}^{D\times\ell}] \in \mathbb{Z}_q^{D\times 2\ell};\\
\mathsf{Ext}_{2m}(\mathbf{y}) \to \hat{\mathbf{y}} \in \mathsf{B}_m^2;\ \mathbf{M}_2' = [\mathbf{M}_2 | \mathbf{0}^{D\times m}] \in \mathbb{Z}_q^{D\times 2m};\\
\mathsf{DecExt}_{2m,p-1}(\mathbf{s}) \to \hat{\mathbf{s}} \in \mathsf{B}_{2m\delta_{p-1}}^3;\ \mathbf{M}_4' = \mathbf{M}_4\cdot\widehat{\mathbf{K}}_{2m,p-1} \in \mathbb{Z}_q^{D\times 6m\delta_{p-1}};\\
\mathsf{DecExt}_{L_0,B}(\mathbf{e}) \to \hat{\mathbf{e}} \in \mathsf{B}_{L_0\delta_B}^3;\ \mathbf{M}_5' = \mathbf{M}_5\cdot\widehat{\mathbf{K}}_{L_0,B} \in \mathbb{Z}_q^{D\times 3L_0\delta_B}.
\end{cases}$$

Now, let $L = 3m\delta_\beta(2\ell+2) + 2\ell + 2m + 2mN + 6m\delta_{p-1} + 3L_0\delta_B$, and construct matrix $\mathbf{P} = \big[\mathbf{F}' | \mathbf{M}_1' | \mathbf{M}_2' | \mathbf{M}_3 | \mathbf{M}_4' | \mathbf{M}_5'\big] \in \mathbb{Z}_q^{D\times L}$ and vector

$$\mathbf{x} = \big(\hat{\mathbf{v}}_1^T \| \hat{\mathbf{v}}_2^T \| \tau[1]\hat{\mathbf{v}}_2^T \| \ldots \| \tau[\ell]\hat{\mathbf{v}}_2^T \| \ldots \| \tau[2\ell]\hat{\mathbf{v}}_2^T \| \hat{\tau}^T \| \hat{\mathbf{y}}^T \| \mathfrak{m}^T \| \hat{\mathbf{s}}^T \| \hat{\mathbf{e}}^T\big)^T,$$

then we will obtain the equation $\mathbf{P}\cdot\mathbf{x} = \mathbf{c} \bmod q$.

Before going on, we define VALID as the set of $\mathbf{w} \in \{-1, 0, 1\}^L$ of the form:

$$\mathbf{w} = \big(\mathbf{w}_1^T \| \mathbf{w}_2^T \| g_1\mathbf{w}_2^T \| \ldots \| g_{2\ell}\mathbf{w}_2^T \| \mathbf{g}^T \| \mathbf{w}_3^T \| \mathbf{w}_4^T \| \mathbf{w}_5^T \| \mathbf{w}_6^T\big)^T$$

for some $\mathbf{w}_1, \mathbf{w}_2 \in \mathsf{B}_{m\delta_\beta}^3$, $\mathbf{g} = (g_1, \ldots, g_{2\ell}) \in \mathsf{B}_{2\ell}$, $\mathbf{w}_3 \in \mathsf{B}_m^2$, $\mathbf{w}_4 \in \mathsf{CorEnc}(mN)$, $\mathbf{w}_5 \in \mathsf{B}_{2m\delta_{p-1}}^3$, and $\mathbf{w}_6 \in \mathsf{B}_{L_0\delta_B}^3$. It can be checked that the constructed vector $\mathbf{x}$ belongs to this tailored set VALID.

Step 2: *Specifying the set $\mathcal{S}$ and permutations of L elements $\{T_\pi : \pi \in \mathcal{S}\}$ for which the conditions in (13) hold.*

– Define $\mathcal{S} = \mathcal{S}_{3m\delta_\beta} \times \mathcal{S}_{3m\delta_\beta} \times \mathcal{S}_{2\ell} \times \mathcal{S}_{2m} \times \{0,1\}^{mN} \times \mathcal{S}_{6m\delta_{p-1}} \times \mathcal{S}_{3L_0\delta_B}$.
– For $\pi = (\phi, \psi, \gamma, \rho, \mathbf{b}, \eta, \xi) \in \mathcal{S}$ and $\mathbf{z} = \big(\mathbf{z}_0^1 \| \mathbf{z}_0^2 \| \mathbf{z}_1 \| \ldots \| \mathbf{z}_{2\ell} \| \mathbf{g} \| \mathbf{t}_1 \| \mathbf{t}_2 \| \mathbf{t}_3 \| \mathbf{t}_4\big) \in \mathbb{Z}_q^L$, where $\mathbf{z}_0^1, \mathbf{z}_0^2, \mathbf{z}_1, \ldots, \mathbf{z}_{2\ell} \in \mathbb{Z}_q^{3m\delta_\beta}$, $\mathbf{g} \in \mathbb{Z}_q^{2\ell}$, $\mathbf{t}_1 \in \mathbb{Z}_q^{2m}$, $\mathbf{t}_2 \in \mathbb{Z}_q^{2mN}$, $\mathbf{t}_3 \in \mathbb{Z}_q^{6m\delta_{p-1}}$, and $\mathbf{t}_4 \in \mathbb{Z}_q^{3L_0\delta_B}$, we define:

$$T_\pi(\mathbf{z}) = \big(\phi(\mathbf{z}_0^1)^T \| \psi(\mathbf{z}_0^2)^T \| \psi(\mathbf{z}_{\gamma(1)})^T \| \ldots \| \psi(\mathbf{z}_{\gamma(2\ell)})^T \| \gamma(\mathbf{g})^T \| \rho(\mathbf{t}_1)^T \| E_{\mathbf{b}}(\mathbf{t}_2)^T \| \eta(\mathbf{t}_3)^T \| \xi(\mathbf{t}_4)^T \big)^T$$

as the permutation that transforms $\mathbf{z}$ as follows:
1. It rearranges the order of the 2ℓ blocks $\mathbf{z}_1, \ldots, \mathbf{z}_{2\ell}$ according to γ.
2. It then permutes block $\mathbf{z}_0^1$ according to ϕ, blocks $\mathbf{z}_0^2$, $\{\mathbf{z}_i\}_{i=1}^{2\ell}$ according to ψ, block $\mathbf{g}$ according to γ, block $\mathbf{t}_1$ according to ρ, block $\mathbf{t}_2$ according to $E_{\mathbf{b}}$, block $\mathbf{t}_3$ according to η, and block $\mathbf{t}_4$ according to ξ.

It can be check that (13) holds. Therefore, we can obtain a statistical ZKAoK for the given relation by running the protocol in Section 5.1.

5.5 The Underlying ZKAoK for the Group Signature Scheme

The argument system upon which our group signature scheme is built can be summarized as follows.

Common Input: Matrices $\mathbf{A}, \{\mathbf{A}_j\}_{j=0}^{\ell}, \mathbf{B} \in \mathbb{Z}_q^{n\times m}$, $\mathbf{D}_0, \mathbf{D}_1 \in \mathbb{Z}_q^{2n\times 2m}$, $\mathbf{F} \in \mathbb{Z}_q^{4n\times 4m}$, $\mathbf{H}_{2n\times m} \in \mathbb{Z}_q^{2n\times m}$, $\mathbf{H}_{4n\times 2m} \in \mathbb{Z}_q^{4n\times 2m}$, $\mathbf{G}_0 \in \mathbb{Z}_q^{n\times 2m}$; vectors $\mathbf{u} \in \mathbb{Z}_q^n$, $\mathbf{c}_1 \in \mathbb{Z}_q^m$, $\mathbf{c}_2 \in \mathbb{Z}_q^{2m}$.

Prover's Input: $\mathbf{z} \in [-\beta, \beta]^{4m}$, $\mathbf{y} \in \{0,1\}^{2m}$, $\mathbf{w} \in \{0,1\}^m$, $\mathbf{d}_1, \mathbf{d}_2 \in [-\beta, \beta]^m$, $\mathbf{s} \in [-\beta, \beta]^{2m}$, $\mathrm{id} = (\mathrm{id}[1], \ldots, \mathrm{id}[\ell])^T \in \{0,1\}^\ell$, $\mathbf{e}_0 \in [-B, B]^n$, $\mathbf{e}_1 \in [-B, B]^m$, $\mathbf{e}_2 \in [-B, B]^{2m}$.

Prover's Goal: Convince the verifier in ZK that

$$\begin{cases} \mathbf{F} \cdot \mathbf{z} = \mathbf{H}_{4n\times 2m} \cdot \mathbf{y} \bmod q; \;\; \mathbf{H}_{2n\times m} \cdot \mathbf{w} = \mathbf{D}_0 \cdot \mathbf{y} + \mathbf{D}_1 \cdot \mathbf{s} \bmod q; \\ \mathbf{A} \cdot \mathbf{d}_1 + \mathbf{A}_0 \cdot \mathbf{d}_2 + \sum_{j=1}^{\ell} \mathbf{A}_j \cdot (\mathrm{id}[j] \cdot \mathbf{d}_2) - \mathbf{D} \cdot \mathbf{w} = \mathbf{u} \bmod q; \\ \mathbf{c}_1 = \mathbf{B}^T \cdot \mathbf{e}_0 + \mathbf{e}_1 \bmod q; \;\; \mathbf{c}_2 = \mathbf{G}_0^T \cdot \mathbf{e}_0 + \mathbf{e}_2 + \lfloor q/2 \rfloor \cdot \mathbf{y} \bmod q. \end{cases}$$

Using the same strategy as in Sections 5.3 and 5.4, we can derive a statistical ZKAoK for the above relation from the protocol in Section 5.1. As the transformations are similar to those in Section 5.4, we only sketch main points.

In the first step, we combine the given equations to an equation of the form:

$$\mathbf{M} \cdot \begin{pmatrix} \mathbf{d}_1 \\ \mathbf{s} \\ \mathbf{z} \end{pmatrix} + \mathbf{M}_0 \cdot \mathbf{d}_2 + \sum_{j=1}^{\ell} \mathbf{M}_j(\mathrm{id}[j]\mathbf{d}_2) + \mathbf{M}' \cdot \begin{pmatrix} \mathbf{w} \\ \mathbf{y} \end{pmatrix} + \mathbf{M}'' \cdot \begin{pmatrix} \mathbf{e}_0 \\ \mathbf{e}_1 \\ \mathbf{e}_2 \end{pmatrix} = \mathbf{v} \bmod q,$$

where matrices $\mathbf{M}, \mathbf{M}_0, \ldots, \mathbf{M}_\ell, \mathbf{M}', \mathbf{M}''$ and vector $\mathbf{v}$ are built from the input.

We then apply the techniques of Section 5.2 for $\mathbf{x}_0 = (\mathbf{d}_1^T \| \mathbf{s}^T \| \mathbf{z}^T)^T \in [-\beta, \beta]^{7m}$, $\mathbf{d}_2 \in [-\beta, \beta]^m$; $\mathbf{x}_1 = (\mathbf{w}^T \| \mathbf{y}^T)^T \in \{0,1\}^{3m}$; and $\mathbf{x}_2 = (\mathbf{e}_0^T \| \mathbf{e}_1^T \| \mathbf{e}_2^T)^T \in [-B, B]^{n+3m}$. This allows us to obtain a unified equation $\mathbf{P} \cdot \mathbf{x} = \mathbf{v} \bmod q$, and to define the sets VALID, $\mathcal{S}$, and permutations $\{T_\pi : \pi \in \mathcal{S}\}$ so that the conditions in (13) hold, in a similar manner as in Section 5.4.

Acknowledgements

We thank Damien Stehlé for useful discussions. The first author was funded by the "Programme Avenir Lyon Saint-Etienne de l'Université de Lyon" in the framework of the programme "Investissements d'Avenir" (ANR-11-IDEX-0007). San Ling, Khoa Nguyen and Huaxiong Wang were supported by the "Singapore Ministry of Education under Research Grant MOE2013-T2-1-041". Huaxiong Wang was also supported by NTU under Tier 1 grant RG143/14.

References

1. M. Abe, M. Chase, B. David, M. Kohlweiss, R. Nishimaki, and M. Ohkubo. Constant-size structure-preserving signatures: Generic constructions and simple assumptions. In *ASIACRYPT 2012*, pages 4–24. Springer, 2012.
2. M. Abe, G. Fuchsbauer, J. Groth, K. Haralambiev, and M. Ohkubo. Structure-preserving signatures and commitments to group elements. In *CRYPTO 2010*, pages 209–236. Springer, 2010.
3. S. Agrawal, D. Boneh, and X. Boyen. Efficient lattice (H)IBE in the standard model. In *EUROCRYPT 2010*, pages 553–572. Springer, 2010.
4. J. Alwen and C. Peikert. Generating shorter bases for hard random lattices. In *STACS 2009*, pages 75–86. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany, 2009.
5. G. Asharov, A. Jain, A. Lopez-Alt, E. Tromer, V. Vaikuntanathan, and D. Wichs. Multiparty computation with low communication, computation and interaction via threshold FHE. In *EUROCRYPT 2012*, pages 483–501. Springer, 2012.
6. G. Ateniese, J. Camenisch, M. Joye, and G. Tsudik. A practical and provably secure coalition-resistant group signature scheme. In *CRYPTO 2000*, pages 255–270. Springer, 2000.
7. S. Bai, A. Langlois, T. Lepoint, D. Stehlé, and R. Steinfeld. Improved security proofs in lattice-based cryptography: Using the Rényi divergence rather than the statistical distance. In *ASIACRYPT 2015*. Springer, 2015.
8. W. Banaszczyk. New bounds in some transference theorems in the geometry of number. *Math. Ann.*, 296:625–635, 1993.
9. M. Belenkiy, J. Camenisch, M. Chase, M. Kohlweiss, A. Lysyanskaya, and H. Shacham. Randomizable proofs and delegatable anonymous credentials. In *CRYPTO 2009*, pages 108–125. Springer, 2009.
10. M. Belenkiy, M. Chase, M. Kohlweiss, and A. Lysyanskaya. P-signatures and noninteractive anonymous credentials. In *TCC 2008*. Springer, 2008.
11. M. Bellare, D. Micciancio, and B. Warinschi. Foundations of group signatures: Formal definitions, simplified requirements, and a construction based on general assumptions. In *EUROCRYPT 2003*. Springer, 2003.
12. M. Bellare and P. Rogaway. Random oracles are practical: a paradigm for designing efficient protocols. In *ACM-CCS 1993*, pages 62–73. ACM, 1993.
13. M. Bellare, H. Shi, and C. Zhang. Foundations of group signatures: The case of dynamic groups. In *CT-RSA 2005*. Springer, 2005.
14. F. Benhamouda, J. Camenisch, S. Krenn, V. Lyubashevsky, and G. Neven. Better zero-knowledge proofs for lattice encryption and their application to group signatures. In *ASIACRYPT 2014*, number 8873, 2014.

15. F. Benhamouda, S. Krenn, V. Lyubashevsky, and K. Pietrzak. Efficient zero-knowledge proofs for commitments from learning with errors over rings. In *ESORICS 2015*, pages 305–325. Springer, 2015.
16. F. Böhl, D. Hofheinz, T. Jager, J. Koch, and C. Striecks. Confined guessing: New signatures from standard assumptions. *Journal of Cryptology*, 28(1):176–208, 2015.
17. D. Boneh and X. Boyen. Efficient selective-ID secure identity-based encryption without random oracles. In *EUROCRYPT 2004*, pages 223–238. Springer, 2004.
18. D. Boneh, X. Boyen, and H. Shacham. Short group signatures. In *CRYPTO 2004*, pages 41–55. Springer, 2004.
19. X. Boyen. Lattice mixing and vanishing trapdoors: A framework for fully secure short signatures and more. In *PKC 2010*, pages 499–517. Springer, 2010.
20. X. Boyen and B. Waters. Compact group signatures without random oracles. In *EUROCRYPT 2006*, pages 427–444. Springer, 2006.
21. X. Boyen and B. Waters. Full-domain subgroup hiding and constant-size group signatures. In *PKC 2007*, pages 1–15. Springer, 2007.
22. Z. Brakerski and Y. T. Kalai. A framework for efficient signatures, ring signatures and identity based encryption in the standard model. *IACR Cryptology ePrint Archive*, 2010:86, 2010.
23. Z. Brakerski, A. Langlois, C. Peikert, O. Regev, and D. Stehlé. On the classical hardness of learning with errors. In *STOC 2013*, pages 575–584. ACM, 2013.
24. J. Camenisch and T. Gross. Efficient attributes for anonymous credentials. In *ACM-CCS 2008*, pages 345–356. ACM, 2008.
25. J. Camenisch, S. Hohenberger, and A. Lysyanskaya. Compact e-cash. In *EUROCRYPT 2005*, pages 302–321. Springer, 2005.
26. J. Camenisch, A. Kiayias, and M. Yung. On the portability of generalized Schnorr proofs. In *EUROCRYPT 2009*, pages 425–442. Springer, 2009.
27. J. Camenisch, S. Krenn, A. Lehmann, G.-L. Mikkelsen, G. Neven, and M.-. Pedersen. Formal treatment of privacy-enhancing credential systems. In *SAC 2015*, pages 3–24. Springer, 2015.
28. J. Camenisch and A. Lysyanskaya. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In *EUROCRYPT 2001*, pages 93–118. Springer, 2001.
29. J. Camenisch and A. Lysyanskaya. Dynamic accumulators and application to efficient revocation of anonymous credentials. In *CRYPTO 2002*, pages 61–76. Springer, 2002.
30. J. Camenisch and A. Lysyanskaya. A signature scheme with efficient protocols. In *SCN 2002*, pages 268–289. Springer, 2002.
31. J. Camenisch and A. Lysyanskaya. Signature schemes and anonymous credentials from bilinear maps. In *CRYPTO 2004*, pages 56–72. Springer, 2004.
32. J. Camenisch, G. Neven, and M. Rückert. Fully anonymous attribute tokens from lattices. In *SCN 2012*, pages 57–75. Springer, 2012.
33. D. Cash, D. Hofheinz, E. Kiltz, and C. Peikert. Bonsai trees, or how to delegate a lattice basis. In *EUROCRYPT 2010*, pages 523–552. Springer, 2010.
34. D. Chaum. Security without identification: Transactions ssystem to make big brother obsolete. *Communications of the ACM*, 28(10):1030–1044, 1985.
35. D. Chaum and E. Van Heyst. Group signatures. In *EUROCRYPT 1991*, pages 257–265. Springer, 1991.
36. I. Damgård. Efficient concurrent zero-knowledge in the auxiliary string model. In *EUROCRYPT 2000*, pages 418–430. Springer, 2000.
37. I. Damgård, D. Hofheinz, E. Kiltz, and R. Thorbek. Public-key encryption with non-interactive opening. In *CT-RSA 2008*, pages 239–255. Springer, 2008.

38. C. Delerablée and D. Pointcheval. Dynamic fully anonymous short group signatures. In *VIETCRYPT 2006*, volume 4341 of *LNCS*, pages 193–210. Springer, 2006.
39. M. F. Ezerman, H. T. Lee, S. Ling, K. Nguyen, and H. Wang. A provably secure group signature scheme from code-based assumptions. In *ASIACRYPT 2015*, pages 260–285. Springer, 2015.
40. A. Fiat and A. Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *CRYPTO 1986*, pages 186–194. Springer, 1987.
41. C. Gentry. Fully homomorphic encryption using ideal lattices. In *STOC 2009*, pages 169–178. ACM, 2009.
42. C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *STOC 2008*, pages 197–206. ACM, 2008.
43. S. Goldwasser, S. Micali, and C. Rackoff. The knowledge complexity of interactive proof-systems. In *STOC 1985*, pages 291–304. ACM, 1985.
44. S. Gorbunov, V. Vaikuntanathan, and H. Wee. Attribute-based encryption for circuits. In *STOC 2013*, pages 545–554. ACM, 2013.
45. S. Gorbunov, V. Vaikuntanathan, and H. Wee. Predicate encryption for circuits from LWE. In *CRYPTO 2015*, number 9216, pages 503–523. Springer, 2015.
46. S. D. Gordon, J. Katz, and V. Vaikuntanathan. A group signature scheme from lattice assumptions. In *ASIACRYPT 2010*, pages 395–412. Springer, 2010.
47. M. Green and S. Hohenberger. Universally composable adaptive oblivious transfer. In *ASIACRYPT 2008*, pages 179–197. Springer, 2008.
48. J. Groth. Fully anonymous group signatures without random oracles. In *ASIACRYPT 2007*, pages 164–180. Springer, 2007.
49. J. Groth and A. Sahai. Efficient non-interactive proof systems for bilinear groups. In *EUROCRYPT 2008*, pages 415–432. Springer, 2008.
50. S. Hohenberger and B. Waters. Short and stateless signatures from the RSA assumption. In *CRYPTO 2009*, pages 654–670. Springer, 2009.
51. A. Jain, S. Krenn, K. Pietrzak, and A. Tentes. Commitments and efficient zero-knowledge proofs from learning parity with noise. In *ASIACRYPT 2012*, pages 663–680. Springer, 2012.
52. A. Kawachi, K. Tanaka, and K. Xagawa. Concurrently secure identification schemes based on the worst-case hardness of lattice problems. In *ASIACRYPT 2008*, pages 372–389. Springer, 2008.
53. A. Kiayias, Y. Tsiounis, and M. Yung. Group encryption. In *ASIACRYPT 2007*, pages 181–199. Springer, 2007.
54. A. Kiayias and M. Yung. Group signatures with efficient concurrent join. In *EUROCRYPT 2005*, pages 198–214. Springer, 2005.
55. A. Kiayias and M. Yung. Secure scalable group signature with dynamic joins and separable authorities. *Int. Journal of Security and Networks*, 1(1):24–45, 2006.
56. F. Laguillaumie, A. Langlois, B. Libert, and D. Stehlé. Lattice-based group signatures with logarithmic signature size. In *ASIACRYPT 2013*, pages 41–61. Springer, 2013.
57. A. Langlois, S. Ling, K. Nguyen, and H. Wang. Lattice-based group signature scheme with verifier-local revocation. In *PKC 2014*, pages 345–361. Springer, 2014.
58. A. Langlois, D. Stehlé, and R. Steinfeld. GGHLite: More efficient multilinear maps from ideal lattices. In *EUROCRYPT 2014*, pages 239–256. Springer, 2014.
59. B. Libert, S. Ling, F. Mouhartem, K. Nguyen, and H. Wang. Signature schemes with efficient protocols and dynamic group signatures from lattice assumptions. Cryptology ePrint Archive: Report 2016/101, 2016.

60. B. Libert, S. Ling, K. Nguyen, and H. Wang. Zero-knowledge arguments for lattice-based accumulators: Logarithmic-size ring signatures and group signatures without trapdoors. In *EUROCRYPT 2016*, pages 1–31. Springer, 2016.
61. S. Ling, K. Nguyen, D. Stehlé, and H. Wang. Improved zero-knowledge proofs of knowledge for the ISIS problem, and applications. In *PKC 2013*, pages 107–124. Springer, 2013.
62. S. Ling, K. Nguyen, and H. Wang. Group signatures from lattices: Simpler, tighter, shorter, ring-based. In *PKC 2015*, pages 427–449. Springer, 2015.
63. V. Lyubashevsky. Lattice-based identification schemes secure under active attacks. In *PKC 2008*, pages 162–179. Springer, 2008.
64. V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. In *EUROCRYPT 2010*, pages 1–23. Springer, 2010.
65. D. Micciancio and C. Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In *EUROCRYPT 2012*, pages 700–718. Springer, 2012.
66. P. Q. Nguyen, J. Zhang, and Z. Zhang. Simpler efficient group signatures from lattices. In *PKC 2015*, pages 401–426. Springer, 2015.
67. C. Papamanthou, E. Shi, R. Tamassia, and K. Yi. Streaming authenticated data structures. In *EUROCRYPT 2013*, pages 353–370. Springer, 2013.
68. C. Peikert. Public-key cryptosystems from the worst-case shortest vector problem. In *STOC 2009*, pages 333–342. ACM, 2009.
69. O. Regev. On lattices, learning with errors, random linear codes, and cryptography. In *STOC 2005*, pages 84–93. ACM, 2005.
70. R. L. Rivest, A. Shamir, and Y. Tauman. How to leak a secret. In *ASIACRYPT 2001*, pages 552–565. Springer, 2001.
71. Y. Sakai, J. Schuldt, K. Emura, G. Hanaoka, and K. Ohta. On the security of dynamic group signatures: Preventing signature hijacking. In *PKC 2012*, pages 715–732. Springer, 2012.
72. J. Stern. A new paradigm for public key identification. *IEEE Transactions on Information Theory*, 42(6):1757–1768, 1996.
73. X. Xie, R. Xue, and M. Wang. Zero knowledge proofs from Ring-LWE. In *CANS 2013*, pages 57–73. Springer, 2013.