

# Lattice-Based Zero-Knowledge Proofs and Applications: Shorter, Simpler, and More General

Vadim Lyubashevsky<sup>1</sup>, Ngoc Khanh Nguyen<sup>1,2</sup>, and Maxime Plançon<sup>1,2</sup>

<sup>1</sup> IBM Research Europe, Zurich

<sup>2</sup> ETH Zurich, Zurich

**Abstract.** We present a much-improved practical protocol, based on the hardness of Module-SIS and Module-LWE problems, for proving knowledge of a short vector  $\vec{s}$  satisfying  $A\vec{s} = \vec{t} \bmod q$ . The currently most-efficient technique for constructing such a proof works by showing that the  $\ell_\infty$  norm of  $\vec{s}$  is small. It creates a commitment to a polynomial vector  $\mathbf{m}$  whose CRT coefficients are the coefficients of  $\vec{s}$  and then shows that (1)  $A \cdot \text{CRT}(\mathbf{m}) = \vec{t} \bmod q$  and (2) in the case that we want to prove that the  $\ell_\infty$  norm is at most 1, the polynomial product  $(\mathbf{m} - \mathbf{1}) \cdot \mathbf{m} \cdot (\mathbf{m} + \mathbf{1})$  equals to 0. While these schemes are already quite good for practical applications, the requirement of using the CRT embedding and only being naturally adapted to proving the  $\ell_\infty$ -norm, somewhat hinders the efficiency of this approach.

In this work, we show that there is a more direct and more efficient way to prove that the coefficients of  $\vec{s}$  have a small  $\ell_2$  norm which does not require an equivocation with the  $\ell_\infty$  norm, nor any conversion to the CRT representation. We observe that the inner product between two vectors  $\vec{r}$  and  $\vec{s}$  can be made to appear as a coefficient of a product (or sum of products) between polynomials which are functions of  $\vec{r}$  and  $\vec{s}$ . Thus, by using a polynomial product proof system and hiding all but one coefficient, we are able to prove knowledge of the inner product of two vectors (or of a vector with itself) modulo  $q$ . Using a cheap, approximate range proof, one can then lift the proof to be over  $\mathbb{Z}$  instead of  $\mathbb{Z}_q$ . Our protocols for proving short norms work over all (interesting) polynomial rings, but are particularly efficient for rings like  $\mathbb{Z}[X]/(X^n + 1)$  in which the function relating the inner product of vectors and polynomial products happens to be a “nice” automorphism.

The new proof system can be plugged into constructions of various lattice-based privacy primitives in a black-box manner. As examples, we instantiate a verifiable encryption scheme and a group signature scheme which are more than twice as compact as the previously best solutions.

## 1 Introduction

The fundamental hardness assumption upon which lattice-based cryptography rests is that it is computationally difficult to find a low-norm vector  $\mathbf{s}$  satisfying

$$\mathbf{A}\mathbf{s} = \mathbf{t} \bmod q. \tag{1}$$

It is then natural that for creating privacy-preserving protocols based on the hardness of lattice problems, one is usually required to prove the knowledge of an  $\mathbf{s}$  satisfying the above, or a related, equality. Unlike in the analogous case of discrete logarithms, where proving knowledge of a secret  $s$  satisfying  $g^s = t$  turns out to have a very simple and efficient solution [40], the added requirement of showing that  $\|\mathbf{s}\|$  is small turns out to be a major complication for *practical* lattice cryptography.

Over polynomial rings (i.e. rings of the form  $\mathbb{Z}[X]/(f(X))$ , where  $f(X)$  is a monic, irreducible polynomial), one can give a fairly-efficient zero-knowledge proof of knowledge of a vector  $\bar{\mathbf{s}}$  and a polynomial  $c$  with small coefficients satisfying

$$\mathbf{A}\bar{\mathbf{s}} = c\mathbf{t} \bmod q, \tag{2}$$

where  $\|\bar{\mathbf{s}}\|$  is some factor (depending on the dimension of  $\mathbf{s}$ ) larger than  $\|\mathbf{s}\|$  [28, 29]. While such proofs are good enough for constructing fairly efficient basic protocols (e.g. signature schemes [28, 29, ?, 18]), the fact that the norm of the extracted  $\bar{\mathbf{s}}$  is noticeably larger than that of  $\mathbf{s}$ , along with the presence of the extra multiplicand  $c$ , makes these proofs awkward to use in many other situations. This very often results in the

protocols employing these proofs being less efficient than necessary, or in not giving the resulting scheme the desired functionality.

As simple examples of inefficiencies that may creep up when only being able to prove (2), consider Regev-style lattice-based encryption schemes (e.g. [39, 36]) where  $\mathbf{s}$  is the randomness (including the message) and  $\mathbf{t}$  is the ciphertext. In order to decrypt, it is necessary for  $\mathbf{t}$  to have a short pre-image, and so being able to only prove knowledge of (2) is not enough to guarantee that the ciphertext  $\mathbf{t}$  can be decrypted because it is  $c\mathbf{t}$  that has a short pre-image, not  $\mathbf{t}$  (and  $c$  is not known to the decryptor). A consequence of this is that the currently most-efficient lattice-based verifiable encryption scheme [31] has the undesirable property that the expected decryption time is equal to the adversary’s running time because the decryptor needs to essentially guess  $c$ . Employing this scheme in the real world would thus require setting up a scenario where the adversary cannot use too much time to construct the proof. Other lattice-based constructions (e.g. group signature schemes [32]) were required to select much larger parameters than needed in order to accommodate the presence of the multiplicand  $c$  and the “slack” between the length of the known solution  $\mathbf{s}$  and the solution  $\bar{\mathbf{s}}$  that one can prove.

### 1.1 Prior Art for Proofs of (1)

Early protocols for exactly proving (1) used the combinatorial algorithm of Stern [41] to prove that the  $\ell_\infty$  norm of  $\mathbf{s}$  is bounded by revealing a random permutation of  $\mathbf{s}$ . The main problem with these protocols was that their soundness error was  $2/3$ , and so they had to be repeated around 200 times to achieve an acceptably small (i.e.  $2^{-128}$ ) soundness error. This resulted in proofs for even basic statements<sup>3</sup> being more than 1MB in size [27], while more interesting constructions required outputs on the order of dozens of Megabytes (e.g. [26]). A noticeable improvement was achieved in [10] by generically combining Stern’s protocol with a “cut-and-choose” technique to noticeably decrease the soundness error of each protocol run (at the expense of higher running times). This allowed proofs for basic statements to be around 200KB in size.

A very different, more algebraic, approach for proving (1) utilized lattice-based commitments and zero-knowledge proofs about committed values to prove relations between the coefficients of  $\mathbf{s}$  and also prove a bound on its  $\ell_\infty$  norm. The first such protocols [42, 12, 20] had proof sizes that were on the order of several hundred kilobytes. These schemes were greatly improved in [4, 19], where it was shown how to very efficiently prove products of polynomial products over a ring and then linear relations over the CRT coefficients of committed values. Optimizations of these techniques [35] decreased the proof size for the basic example to around 33KB.

The high level idea for these proofs, when  $\mathbf{s}$  has coefficients in the set  $\{-1, 0, 1\}$ , is to create a BDLOP commitment [6] to a polynomial  $\mathbf{m}$  whose CRT coefficients are the coefficients of  $\mathbf{s}$ , prove this (linear) relationship as well as the one in (1) [19], and then prove that  $(\mathbf{m} - \mathbf{1}) \cdot \mathbf{m} \cdot (\mathbf{m} + \mathbf{1}) = 0$  [4].

There are a few intrinsic elements of this approach which hinder its efficiency, especially in certain situations. The first is that the CRT of  $\mathbf{m}$  consists of large coefficients, and so committing to it requires using a more expensive commitment scheme, which is especially costly when  $\mathbf{s}$  is long<sup>4</sup> (we discuss this in more detail when talking about various commitments in Section 1.3). Another downside is that for vectors  $\mathbf{s}$  with somewhat-large coefficients, such as ones that are obtained from trapdoor sampling (e.g. [1, 38]), proving the smallness of the  $\ell_\infty$ -norm becomes significantly costlier because the degree of the polynomial product increases. There is also an incompatibility between the requirement that the underlying ring has a lot of CRT slots and negligible soundness error of the protocol – thus a part of the protocol needs to be repeated for soundness amplification. And finally, proving the  $\ell_2$  norm, rather than the  $\ell_\infty$  one, is very often what one would like to do when constructing proofs for lattice-based primitives. This is because efficient trapdoor-sampling used in many lattice primitives produces vectors of (tightly) bounded  $\ell_2$  norm, and noise also generation generally results in tight  $\ell_2$ -norm bounds.

<sup>3</sup> A standard example that has been used for comparison-purposes in several works is  $1024 \times 2048$  integer matrix  $\mathbf{A}$ , a 32-bit modulus  $q$ , and  $\mathbf{s}$  having coefficients in  $\{-1, 0, 1\}$  (or  $\|\mathbf{s}\| \leq \sqrt{2048}$ ).

<sup>4</sup> The aforementioned framework was most appropriate for committing to small-dimensional messages (e.g. in protocols related to anonymous transactions (e.g. [22, 35, 21]) and proving various relationships between them.

## 1.2 Our Results

We propose a simpler, more efficient, and more direct approach for proving a tight bound on the  $\ell_2$  norm of  $\mathbf{s}$  satisfying (1). Unlike in the previous approach, we do not need to recommit to  $\mathbf{s}$  in CRT form, and therefore don't have a ring algebra requirement which had a negative effect on the protocol soundness. Furthermore, not needing to create a BDLOP commitment to  $\mathbf{s}$  noticeably shrinks the proof size. In particular, we define a commitment scheme which combines the Ajtai [2] and BDLOP [6] commitments into one, and then put the long commitment to  $\mathbf{s}$  into the "Ajtai" part of the commitment scheme, which does not increase the commitment size.<sup>5</sup>

We then observe that the inner product of two vectors over  $\mathbb{Z}$  can be made to appear as the constant coefficient of a polynomial product, or as a coefficient in a sum of polynomial products. Our protocol for proving the  $\ell_2$ -norm of  $\mathbf{s}$  is then a specific application of a more general protocol that can prove knowledge of constant coefficients of quadratic relations over polynomial rings for messages that are committed in the "Ajtai" and "BDLOP" parts of our new commitment. Our protocols are built up in a black-box manner from basic building blocks, and can then also be used in a black box manner for implementing the zero-knowledge proof parts of various lattice-based primitives. As examples, the ZK proof of the basic relation from (1) is  $\approx 2.5X$  shorter than in previous works, a verifiable encryption scheme can be as short as the one from [31] without the constraint that the decryption time is proportional to the adversary's attack time, and we give a group signature scheme whose signatures are more than  $2X$  smaller than the currently most compact one.

Our proof system for the basic equality from (1) is around 14KB, and approximately 8KB of that consists of just the "minimum" commitment (i.e. a commitment to just one element in  $\mathcal{R}_q$  that doesn't include  $\mathbf{s}$ ) and its opening proof. This shows that our construction is quite close to being optimal for any approach that requires creating a commitment to  $\mathbf{s}$  using known lattice-based commitment schemes. Since all zero-knowledge proofs that we're aware of for showing that a secret  $s$  satisfies  $f(s)$  work by first committing to  $s$ , it appears that any significant improvement to this proof system (e.g. another factor of 2) would require noticeable improvements in fundamental lattice primitives, basing security on stronger assumptions, or a noticeable departure from the current approach.

We now give a detailed overview of the techniques and results in this work, and then sketch how our framework can be used to construct lattice-based privacy protocols.

## 1.3 Techniques Overview

Throughout most of the introduction and paper, we will concentrate on the ring  $\mathcal{R}_q = \mathbb{Z}_q[X]/(X^d+1)$ , as our constructions are most efficient here because they can utilize a specific automorphism in this ring. Towards the end of this section and in Section 7, we describe how to adapt our construction, and most applications, to other rings that do not have this algebraic structure. All our constructions will be based on the hardness of the Module-SIS and Module-LWE problems and one should think of the degree of the underlying ring  $d$  to be something small like 64 or 128 (we use 128 for all our instantiations).

**Commitment Schemes.** In the original Ajtai commitment scheme, implicit in [2], one commits to a message  $\mathbf{s}_1$  using randomness  $\mathbf{s}_2$ , where  $\|\mathbf{s}_i\|$  are small, as

$$\mathbf{A}_1\mathbf{s}_1 + \mathbf{A}_2\mathbf{s}_2 = \mathbf{t} \text{ mod } q. \tag{3}$$

It's easy to see that creating a second valid opening ( $\mathbf{s}'_1, \mathbf{s}'_2$ ) for the same commitment value  $\mathbf{t}$  is equivalent to solving the SIS problem over  $\mathcal{R}_q$ , and the hiding aspect of the commitment scheme is based on the indistinguishability of  $(\mathbf{A}_2, \mathbf{A}_2\mathbf{s}_2)$  from uniform. A useful feature of the above commitment scheme is that the dimension of the message  $\mathbf{s}_1$  does not increase the commitment size. And since the hardness of SIS does not really depend on the dimension of the solution, increasing the dimension of  $\mathbf{s}_1$  does not negatively impact the security either. On the other hand, one does need the coefficients of  $\mathbf{s}_1$  to be small.

<sup>5</sup> The BDLOP part of the commitment scheme is then used for low-dimensional auxiliary elements that will need to be committed to later in the protocol.

A different commitment scheme, called the BDLOP scheme [6], commits to a message  $\mathbf{m}$  using randomness  $\mathbf{s}$  as

$$\begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix} \cdot \mathbf{s} + \begin{bmatrix} \mathbf{0} \\ \mathbf{m} \end{bmatrix} = \begin{bmatrix} \mathbf{t}_A \\ \mathbf{t}_B \end{bmatrix} \bmod q, \quad (4)$$

where only the randomness  $\mathbf{s}$  needs to have a small norm. An opening of this commitment is just  $\mathbf{s}$  since it uniquely determines  $\mathbf{m}$ , and so it is again easy to see that two different openings lead to a solution to SIS for the matrix  $\mathbf{A}$ . The hiding property of this commitment is based on the indistinguishability from uniform of  $\left( \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix}, \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix} \cdot \mathbf{s} \right)$ .

This scheme has two advantages and one disadvantage over the one in (3). The disadvantage is that both the commitment size and the opening size grow linearly with the dimension of the message vector  $\mathbf{m}$ . An advantage is that the coefficients of  $\mathbf{m}$  can be arbitrarily large modulo  $q$ . The other advantage is that if one plans ahead and sets the dimension of  $\mathbf{s}$  large enough, one can very cheaply append commitments of new elements in  $\mathcal{R}_q$ . For example, if we have already created a commitment to  $\mathbf{m}$  as in (4) and would like to commit to another polynomial vector  $\mathbf{m}'$ , we can compute  $\mathbf{B}'\mathbf{s} + \mathbf{m}' = \mathbf{t}'_B \bmod q$ , where  $\mathbf{B}'$  is some public

randomness. If  $\left( \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \\ \mathbf{B}' \end{bmatrix}, \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \\ \mathbf{B}' \end{bmatrix} \cdot \mathbf{s} \right)$  is indistinguishable from uniform, then  $(\mathbf{t}_A, \mathbf{t}_B, \mathbf{t}'_B)$  is a commitment to  $\mathbf{m}, \mathbf{m}'$ . Note that committing to  $k$  extra  $\mathcal{R}_q$  elements requires growing the commitment size by only  $k$   $\mathcal{R}_q$  elements, something that cannot be done using the scheme from (3).

For optimality, our construction will require features from both of these schemes, and it actually turns out to be possible to combine the two of them into one. So to commit to a message  $\mathbf{s}_1$  with a small norm, and a message  $\mathbf{m}$  with unrestricted coefficients (modulo  $q$ ), one can create a commitment

$$\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{0} \end{bmatrix} \cdot \mathbf{s}_1 + \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{B} \end{bmatrix} \cdot \mathbf{s}_2 + \begin{bmatrix} \mathbf{0} \\ \mathbf{m} \end{bmatrix} = \begin{bmatrix} \mathbf{t}_A \\ \mathbf{t}_B \end{bmatrix} \bmod q, \quad (5)$$

where the randomness is  $\mathbf{s}_2$ . We will call this combination of the Ajtai and BDLOP commitment scheme, the ABDLOP commitment. The savings over creating two separate commitments is that instead of needing the  $\mathbf{t}$  term from (3) and the  $\mathbf{t}_A$  term from (4), we only have the  $\mathbf{t}_A$  term. So we get an Ajtai commitment to  $\mathbf{s}_1$  for free! And similarly, the opening does not require both  $\mathbf{s}_2$  from (3) and  $\mathbf{s}$  from (4).

One can show that (5) is indeed a commitment scheme and has an efficient zero-knowledge opening proof.<sup>6</sup> Furthermore, there is also an efficient zero-knowledge proof (much like in [6]) which allows one to efficiently show that the committed values  $\mathbf{s}_1, \mathbf{m}$  satisfy a relation over  $\mathcal{R}_q$

$$\mathbf{R}_1 \mathbf{s}_1 + \mathbf{R}_m \mathbf{m} = \mathbf{u} \bmod q, \quad (6)$$

where the matrices  $\mathbf{R}_1, \mathbf{R}_2$ , and the vector  $\mathbf{u}$  are public. This proof system is given in Figure 4, and we just mention that the proof size is not affected by the sizes of the  $\mathbf{R}_i$ . In other words, the proof size for proving linear relations over  $\mathcal{R}_q$  is the same as the proof size of just proving knowledge of the committed values. The only way in which this proof puts a restriction on the underlying ring is that the modulus  $q$  must be large enough so that the extracted SIS solution is hard, and that the challenge set  $\mathcal{C}$  is such that the difference of challenges is (with high probability) invertible. This can be done by choosing the modulus  $q$  in a way that  $X^d + 1$  splits into very few irreducible factors of the form  $X^k - r_i$  modulo  $q$  (or the prime factors of  $q$ ), which in turn implies that all elements of  $\mathcal{R}_q$  with small coefficients are invertible [37].

The way this commitment scheme will be used in our protocols is that we will put high-dimensional messages with small coefficients into  $\mathbf{s}_1$ , while putting small-dimensional values with large coefficients – generally auxiliary “garbage terms” that we will need to commit to during the protocol which aid in proving relations among the elements in  $\mathbf{s}_1$  – into  $\mathbf{m}$ .

<sup>6</sup> As for the Ajtai and BDLOP commitments, the opening needs to be carefully defined because the ZK proof only proves approximate relations as in (2). The details are in Section 3.1.

**Inner Products over  $\mathbb{Z}_q$ .** Suppose that instead of just wanting to prove linear relations over  $\mathcal{R}_q$ , as above, we wanted to prove linear relations over  $\mathbb{Z}_q$ . That is, if we let  $R_1, R_m$  be integer matrices, and we write  $\vec{s}_1$  and  $\vec{m}$  to be integer vectors whose coefficients are the integer coefficients of the polynomial vectors  $\mathbf{s}_1$  and  $\mathbf{m}$ , then we would like to prove that  $R_1\vec{s}_1 + R_m\vec{m} = \vec{u} \pmod q$ .

An important observation is the following: if  $\vec{r} = (r_0, r_1, \dots, r_{d-1}), \vec{s} = (s_0, s_1, \dots, s_{d-1}) \in \mathbb{Z}_q^d$  are vectors and  $r(X) = \sum_i r_i X^i, s(X) = \sum_i s_i X^i \in \mathcal{R}_q$  are the corresponding polynomials, then  $\langle \vec{r}, \vec{s} \rangle \pmod q$  is equal to the constant coefficient of the polynomial product  $r(X^{-1}) \cdot s(X)$  over  $\mathcal{R}_q$ .<sup>7</sup> Similarly, for  $\vec{r}, \vec{s} \in \mathbb{Z}_q^{kd}$ , one can define the corresponding polynomial vectors  $\mathbf{r} = (r_1, \dots, r_k), \mathbf{s} = (s_1, \dots, s_k) \in \mathcal{R}_q^k$  to have the same coefficients as  $\vec{r}, \vec{s}$  in the straightforward manner, then  $\langle \vec{r}, \vec{s} \rangle \pmod q$  is equal to the constant coefficient of  $\sum_i r_i(X^{-1}) \cdot s_i(X)$ , where the multiplication is performed over  $\mathcal{R}_q$ .

For a polynomial  $h = h_0 + h_1X + \dots + h_{d-1}X^{d-1} \in \mathcal{R}_q$ , we will write  $\tilde{h}$  to mean the constant coefficient  $h_0$ . The procedure to prove that  $\langle \vec{r}, \vec{s} \rangle \pmod q = \alpha$  is then to create polynomial vectors  $\mathbf{r}, \mathbf{s}$  such that  $\langle \mathbf{r}, \mathbf{s} \rangle$  (where the inner product is over  $\mathcal{R}_q$ ) is equal to  $\langle \vec{r}, \vec{s} \rangle$ . One can hope to use the protocol from Figure 4 to prove the linear relation over  $\mathcal{R}_q$ , which would imply the linear relation over  $\mathbb{Z}_q$ . The problem is that naively proving the relation over  $\mathcal{R}_q$  would necessarily require the prover to reveal all the coefficients of  $\langle \mathbf{r}, \mathbf{s} \rangle$  instead of just the constant one, which implies giving out extra information about the committed vector  $\vec{s}$ , and so is clearly not zero-knowledge.

We now outline the solution to this problem for general linear functions. For a linear function  $f : \mathcal{R}_q^k \rightarrow \mathcal{R}_q$ , we would like to prove that the committed values  $\mathbf{s}_1, \mathbf{m}$  in the ABDLOP commitment satisfy  $\tilde{f}(\mathbf{s}_1, \mathbf{m}) = 0$  (for aesthetics, we will write  $\tilde{f}(x)$  to mean  $\tilde{f}(x)$ ). In order to mask all but the constant coefficient, we use a masking technique from [19], where the prover first creates a commitment to a polynomial  $g \in \mathcal{R}_q$  such that  $\tilde{g} = 0$  and all of its other coefficients are chosen uniformly at random. In our proof system, he commits to this polynomial in the ‘‘BDLOP part’’ of (5) by outputting  $t_g = \langle \mathbf{b}, \mathbf{s}_2 \rangle + g$ , where  $\mathbf{b}$  is some random public polynomial vector. The verifier then sends a random challenge  $\gamma \in \mathbb{Z}_q$ , and the prover computes

$$h = \gamma \cdot f(\mathbf{s}_1, \mathbf{m}) + g. \quad (7)$$

The prover then creates a proof, as in Figure 4, that the committed values  $\mathbf{s}_1, \mathbf{m}$ , and  $g$  satisfy this linear relation, and sends  $h$  along with this proof to the verifier. The verifier simply checks the validity of the linear proof, and also that  $\tilde{h} = 0 \pmod q$ .

The proof leaks no information about all but the constant coefficient of  $f(\mathbf{s}_1, \mathbf{m})$  because they are masked by the completely random coefficients of  $g$ . To see that this proof is sound, note that for all  $g$ , if  $\tilde{f}(\mathbf{s}_1, \mathbf{m}) \neq 0$ , then  $\Pr_\gamma[\gamma \cdot \tilde{f}(\mathbf{s}_1, \mathbf{m}) + \tilde{g} = 0] \leq 1/q_1$ , where  $q_1$  is the smallest prime factor of  $q$ . In order to reduce the soundness error down to  $\epsilon$ , the prover would need to create a commitment to  $\lambda$  different  $g_i$ , where  $(1/q_1)^\lambda = \epsilon$  and then reply to  $\lambda$  different challenges  $\gamma_i$  by creating  $\lambda$  different  $h_i$  as in (7). Since the  $g_i$  are just one polynomial in  $\mathcal{R}_q$ , the  $h_i$  are also just one polynomial each, and so amplifying the proof requires sending just  $2\lambda$  extra elements in  $\mathcal{R}_q$ .

The above shows that proving one relation  $\tilde{f}(\mathbf{s}_1, \mathbf{m}) = 0$  requires a small number  $\lambda$  of extra polynomials  $g$  and  $h$ . Usually, we will want to prove many such linear equations, and so it would be quite inefficient if our proof size grew linearly in their number. But, just like in the basic protocol in Figure 4, we can show that the number of equations that we need to prove does not affect the size of the proof. If we would like to prove  $k$  equations  $\tilde{f}_i(\mathbf{s}_1, \mathbf{m}) = 0$ , the prover still sends the term  $g$  in the first round (let’s ignore the amplification for now), but this time instead of sending just one random challenge  $\gamma \in \mathbb{Z}_q$ , the verifier sends  $k$  random challenges  $\gamma_i$ . The prover then creates the equation

$$h = \sum_i \gamma_i \cdot f_i(\mathbf{s}_1, \mathbf{m}) + g, \quad (8)$$

<sup>7</sup> For a polynomial  $r(X) = \sum_{i=0}^{d-1} r_i X^i \in \mathcal{R}_q$ ,  $r(X^{-1}) = r_0 - \sum_{i=1}^{d-1} r_i X^{d-i}$ .

and sends  $h$  along with a proof that the  $\mathbf{s}_1, \mathbf{m}$ , and  $g$  satisfy the above. The verifier checks the proof and that  $\tilde{h} = 0 \pmod q$ . The fact that this proof leaks no information and that the soundness error is again  $1/q_1$  is virtually identical as for (7), and we give a full description of this protocol in Figure 5.

**Quadratic Relations and Norms.** In the above, we saw an overview of how one can prove knowledge of inner products over  $\mathcal{R}_q$  and  $\mathbb{Z}_q$  when one of the values is committed to and the other is public. We now show how to do the same thing when both values are in the commitment – in other words, how to prove quadratic relations over committed values.

The most efficient protocol for proving quadratic relations between committed polynomials in  $\mathcal{R}_q$  is given in [4]. That protocol assumes that the elements were committed using the BDLOP commitment scheme, and one can show that a similar approach works for the ABDLOP scheme as well. And so one can prove arbitrary quadratic relations over  $\mathcal{R}_q$  between the committed polynomials in the polynomial vector  $\mathbf{s}_1$  and  $\mathbf{m}$  in (5). We will now explain how to use this proof system, together with the ideas presented above, to construct a proof that the  $\mathbf{s}$  satisfying (1) has small  $\ell_2$ -norm. For simplicity of this description, let's just suppose that we would like to prove that  $\|\mathbf{s}\| = \beta$  instead of  $\|\mathbf{s}\| \leq \beta$ .<sup>8</sup> The idea is to first commit to  $\mathbf{s}$  as part of the  $\mathbf{s}_1$  part of (5) (i.e. in the ‘‘Ajtai part’’ of the ABDLOP scheme). Then we use the observation from the previous section that notes that if  $\mathbf{s}_1 = (s_1, \dots, s_k) \in \mathcal{R}_q^k$ , then  $\|\mathbf{s}\|^2$  is the constant coefficient of  $\sum_i s_i(X^{-1}) \cdot s_i(X)$ . We cannot directly use the proof system for linear proofs because that one assumed that one of the multiplicands was public. We thus need to extend the protocol from [4] to prove knowledge of  $\sum_i s_i(X^{-1}) \cdot s_i(X)$  when having a commitment to  $\mathbf{s}$ .

Let us first recall the idea from [4] and how they can be applied to the ABDLOP commitment. Suppose, for example, that we wanted to prove that  $s_1 s_2 - s_3 = 0$ , and we had commitments to  $s_i$  in the Ajtai part of the ABDLOP commitment (i.e. the  $s_i$  are part of the  $\mathbf{s}_1$  in (5)). If one looks at the protocol in Figure 4 for proving knowledge of committed values in the ABDLOP protocol, then we note that the prover sends the vector  $\mathbf{z}_1 = c\mathbf{s}_1 + \mathbf{y}_1$ . This  $\mathbf{z}_1$  consists of terms  $z_i = s_i c + y_i$ , where  $c$  is a polynomial challenge (with small coefficients) and  $y_i$  is a masking polynomial whose job is to hide  $s_i$ .

The high level idea in which the protocol from [4] (and some that preceded it [12, 20, 42]) proves quadratic relations is by having the verifier create a quadratic equation (in  $c$ ) out of the linear equations  $z_i = cs_i + y_i$ . That is, the verifier computes

$$z_1 z_2 - cz_3 = (s_1 s_2 - s_3)c^2 + g_1 c + g_0, \quad (9)$$

where  $g_1$  and  $g_0$  are some terms which depend on  $y_i$  and  $s_i$  and are committed to by the prover prior to receiving the challenge  $c$ .<sup>9</sup> The above is a quadratic equation in the variable  $c$  (since all the other terms are already committed to), and so if the prover shows that  $z_1 z_2 - cz_3 = g_1 c + g_0$  (i.e. it's actually a linear equation) it will imply that with high probability the quadratic coefficient,  $s_1 s_2 - s_3$  is equal to 0.

To prove that the constant coefficient of  $s(X^{-1}) \cdot s(X)$  is some value  $\beta$ , one can try to do something similar. Here, it becomes important that the function mapping  $s$  to  $s(X^{-1})$  is an automorphism (call it  $\sigma$ ) for  $\mathcal{R}_q$ . Given the term  $z = sc + y$ , the verifier is able to compute

$$\sigma(z) \cdot z - \sigma(c) \cdot c \cdot \beta^2 = (\sigma(s) \cdot s - \beta^2) \cdot \sigma(c) \cdot c + \sigma(s) \cdot y \cdot \sigma(c) + s \cdot \sigma(y) \cdot c + \sigma(y) \cdot y, \quad (10)$$

and, if the above is equal to  $g_2 \cdot \sigma(c) + g_1 \cdot c + g_0$ , would like to conclude that the coefficients in front of  $\sigma(c) \cdot c$  is 0. Unfortunately, we can't conclude this because the  $c$  and  $\sigma(c)$  are not independent. What we instead do is choose the challenges  $c$  from a set that is fixed under this automorphism. Then (10) becomes

$$\sigma(z) \cdot z - c^2 \beta^2 = (\sigma(s) \cdot s - \beta^2) \cdot c^2 + (\sigma(s) \cdot y + s \cdot \sigma(y)) \cdot c + \sigma(y) \cdot y, \quad (11)$$

<sup>8</sup> To prove the latter, one would commit to a vector  $\vec{b}$  which is the binary representation of the integer  $\beta^2 - \|\mathbf{s}\|^2$  and then prove that it is indeed binary and that  $\langle \vec{b}, (1, 2, 2^2, \dots, 0, \dots, 0) \rangle$  is  $\beta^2 - \|\mathbf{s}\|^2$ ; which implies that the latter is positive. Note that it is still a quadratic relation in the committed values  $\mathbf{s}$  and  $\vec{b}$ .

<sup>9</sup> [4] showed that the  $y_i$  were already implicitly committed to by the first part of the protocol.

and we again have a quadratic equation in  $c$ . Luckily, this does not restrict the challenge set too much. In particular, if we choose  $c \in \mathcal{R}_q$  to be of the form  $c = c_0 + \sum_{i=1}^{d/2-1} c_i \cdot (X^i - X^{d-i})$ , where  $c_i \in \mathbb{Z}_q$ , then  $c = \sigma(c)$ .<sup>10</sup> So we are free to set  $d/2$  coefficients of the challenge polynomial instead of the usual  $d$ . So obtaining the same soundness requires the coefficients to be a little larger, but this has a rather small effect on the proof size.

The protocol in Figure 6 is a very general protocol for proving that a quadratic function in the coefficients of  $\mathbf{s}_1$  and  $\mathbf{m}$ , and the automorphisms of  $\mathbf{s}_1$  and  $\mathbf{m}$ , is satisfied as long as the challenge set is fixed under the particular automorphism. If we only want to prove the  $\ell_2$  norm, then we do not want to prove a quadratic function over  $\mathcal{R}_q$ , but rather we just want to prove something about the constant coefficient of a quadratic relation over  $\mathcal{R}_q$ . To do this, we employ the same masking technique as in (7) that we used for our linear proofs over  $\mathbb{Z}_q$ . Furthermore, just like in the linear proofs setting, if we need to prove multiple quadratic relations, we can first combine them into one equation, and then the proof size does not increase. Also note that we can clearly combine linear and quadratic equations together into one quadratic equation. The full protocol is presented in Figure 8.

We are almost done, except for the fact that all of our proofs are modulo  $q$ . That is, the protocol only proves that  $\|\mathbf{s}\|^2 = \beta^2 \pmod q$ , which is not the same as proving  $\|\mathbf{s}\|^2 = \beta^2$ . In order to prove that there is no “wraparound” modulo  $q$ , we employ a version of the “approximate range proof” technique to show that the coefficients of  $\mathbf{s}$  are all small-enough. We do not need a sharp bound on these coefficients, but just need to show that they are small-enough that no wraparound occurs. For this, we use the technique [7, 8, 33, 23] of committing to a masking vector  $\vec{y}$  (in the BDLOP part of (5)), receiving a  $-1/0/1$  challenge matrix  $R$ , and outputting  $\vec{z} = R\vec{s} + \vec{y}$  (and doing a rejection sampling to hide  $\vec{s}$ ). It can be shown that if  $\|\vec{z}\|$  is small, then  $\|\vec{s}\|$  is also small. The dimension of  $\vec{y}$  and  $\vec{z}$  is small (between 128 and 256), and so the extra commitment to  $\vec{y}$  and the revealing of  $\vec{z}$  is inexpensive. The protocol for the approximate range proof is given in Figure 9, and the general protocol proving these approximate range proofs in combination with other quadratic functions is given in Figure 10.

**Putting Everything Together.** The structure for proving (1) involves creating an ABDLOP commitment as in (5) with  $\mathbf{s}_1 = \mathbf{s}$  and making the randomness  $\mathbf{s}_2$  long enough to accommodate future commitments to a few intermediate terms necessary in the proof. One then uses the aforementioned proofs to show that  $\|\mathbf{s}_1\|$  is small, and that the linear equation in (1) is satisfied. Notice that we don’t really need any ring structure on the equation in (1); if it is over  $\mathbb{Z}_q$ , we can simply prove it using the linear proofs over  $\mathbb{Z}_q$ . This is computationally more expensive than if the equation were over  $\mathcal{R}_q$ , because for every inner product over  $\mathbb{Z}_q$ , we have to compute one inner product over  $\mathcal{R}_q$ , but the proof size will be the same.

We also note that the modulus in (1) does not have to be the same as in the commitment scheme. In fact, it will often be necessary to use a larger modulus in the commitment scheme because it has to be larger than  $\|\mathbf{s}\|^2$ . For example, we can set the commitment scheme modulus to  $p \cdot q$  and then simply lift the equation in (1) to this modulus by multiplying both sides of it by  $p$ . As long as the challenge differences are invertible in the ring  $\mathcal{R}_q$  and  $\mathcal{R}_p$ , all the protocols go through unchanged.

Another possibility is, instead of proving  $\mathbf{A}\mathbf{s} = \mathbf{t} \pmod q$ , one proves that

$$\mathbf{A}\mathbf{s} - \mathbf{t} = \mathbf{r} \cdot q \tag{12}$$

over the integers. If each row of  $\mathbf{A}$  consists of  $m$  integer coefficients, then each coefficient of  $\mathbf{r}$  has magnitude at most  $mq$ . One can then do the proof system using a larger modulus  $p$ , and also prove that each coefficient of  $q^{-1}(\mathbf{A}\mathbf{s} - \mathbf{t}) \pmod p$  is small using the approximate range proof. The advantage of this method over using  $pq$  as the modulus for the commitment scheme, as above, is that it allows the commitment scheme modulus  $p$  to be a prime, and so one needs fewer terms for coefficient masking (see the discussion after (7)), which could save a few kilobytes in the complete proof. A disadvantage is that there is now the extra  $\mathbf{r}$  term that needs to be dealt with.

<sup>10</sup> This is easy to see because  $\sigma(X^i - X^{d-i}) = X^{-i} - X^{i-d}$ , and multiplying by  $-X^d = 1$ , we obtain  $\sigma(X^i - X^{d-i}) = -X^{d-i} + X^i$ .

**Extensions.** While we concentrated on proving the smallness of the  $\ell_2$ -norm of a vector  $\vec{s}$  (or more generally the knowledge of the inner product between two vectors), it is also possible to use our techniques to prove many other inter-vector relations. In particular, a useful relation (e.g. if dealing with general functions/circuits) is proving the knowledge of the component-wise product  $\vec{r} \circ \vec{s}$ . This can be generally accomplished by proving a polynomial product over a ring  $\mathcal{R}_p$  of two vectors  $\mathbf{r}$  and  $\mathbf{s}$  whose CRT coefficients are  $\vec{r}$  and  $\vec{s}$ . The important thing is to choose a prime  $p$  such that the polynomial  $X^d + 1$  factors into linear factors modulo  $p$ . As mentioned above, by simply subtracting off the remainder as in (12), one can use different moduli for the commitment scheme for the relations that we would like to prove. Thus one can choose a ‘‘CRT-friendly’’ modulus for the underlying relation, while using a modulus that allows the polynomial differences to be invertible (so not a CRT-friendly one) for the commitment scheme.

We also point out that proving inner products can be directly used to prove another very natural function – showing that all the coefficients of a vector are from the set  $\{0, 1\}$ . For this, one uses the observation that  $\vec{s}$  has coefficients in  $\{0, 1\}$  if and only if  $\langle \vec{s}, \vec{1} - \vec{s} \rangle = 0$ . And since given a commitment for  $\vec{s}$ , one can ‘‘maul it’’ into a commitment to  $\vec{1} - \vec{s}$ , one can generically apply the aforementioned protocol in Figure 8.

**Using Other Rings.** In proving that the norm of a polynomial  $s$  was small, we exploited the fact that in the ring  $\mathcal{R}$ ,  $s(\widetilde{X^{-1}}) \cdot s = \|s\|^2$  and that  $s(\widetilde{X^{-1}})$  was an automorphism. In Section 7, we show that the same high level ideas can also be made to work for rings that don’t have this algebraic structure. Specifically, for all rings  $R = \mathbb{Z}[X]/(X^d + f_{d-1}X^{d-1} + \dots + f_1X \pm 1)$ , there exists a linear function  $g : R \rightarrow R$  such that  $g(\widetilde{r}) \cdot s$  is equal to  $\langle \vec{r}, \vec{s} \rangle$ . If  $g$  is not an automorphism, then proving knowledge of  $\|s\|^2 = g(\widetilde{s}) \cdot s$  would require the prover to commit to both  $s$  and  $g(s)$ , and then also prove the linear relationship between the commitments of  $s$  and  $g(s)$ . Opening two commitments instead of one will increase the proof size, but this is slightly mitigated by the fact that the challenges no longer need to be restricted to be fixed under any automorphism.

**Sample Constructions.** In Section 6, we present various instantiations of lattice-based primitives that can be constructed using our zero-knowledge proof system. We now give a very high-level description of a group signature scheme. In a group signature scheme, the Setup Authority uses a master secret key to distribute member secret keys to the members of the group. The members can then use their secret keys to sign messages on behalf of the group. An entity known as the Opener (or group manager) also has a special secret key that allows him to obtain the identity of the signer of any message. The privacy criterion states that it should be impossible, for everyone but the Opener, to trace back a signature to the particular user, nor link that two signatures were signed by the same user. Conversely, the traceability requirement states that every message signed by a user with identity  $\mu$  will get traced back to him by the Opener. Group signatures are an interesting primitive in their own right, but are particularly useful in determining the practicality of zero-knowledge proofs as they contain some ingredients which are prevalent throughout privacy-based cryptography.

We show how we can use our improved ZK proof to construct a lattice-based group signature following the framework of [16, 32]. The master public key is  $[\mathbf{A} \mid \mathbf{B}]$ ,  $\mathbf{u}$ , and the secret key of a group member with identity  $\mu$  is a short vector  $\begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{bmatrix}$  such that

$$[\mathbf{A} \mid \mathbf{B} + \mu \mathbf{G}] \cdot \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{bmatrix} = \mathbf{u} \bmod q. \quad (13)$$

The setup authority with a trapdoor for the lattice  $\mathcal{L} = \{\mathbf{x} : [\mathbf{A} \mid \mathbf{B}] \cdot \mathbf{x} = \mathbf{0} \bmod q\}$  can create such short vectors which are distributed according to a discrete Gaussian distribution [1, 38].

The group member’s signature of a message consists of a Module-LWE encryption of his identity  $\mu$  as

$$\begin{bmatrix} \mathbf{A}' \\ \mathbf{b} \end{bmatrix} \cdot \mathbf{r} + \begin{bmatrix} 0 \\ [p/2]\mu \end{bmatrix} = \mathbf{t} \bmod p, \quad (14)$$



where  $\mathbf{A}', \mathbf{b}$  is the public key (of the Opener) and  $\mathbf{r}$  is the randomness, together with a ZKPoK that he knows  $\mu, \mathbf{r}$ , and  $\begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{bmatrix}$  satisfying (13) and (14). The message that the user is signing is, as usual, put into the input of the hash function used in the Fiat-Shamir transform of the ZKPoK.

To create this signature, the user commits to  $\mathbf{s}_1, \mathbf{s}_2, \mathbf{r}, \mu$  in the ‘‘Ajtai’’ part of the ABDLOP commitment (5). He then proves that the norms of  $\mathbf{s}_1, \mathbf{s}_2, \mathbf{r}$  are small, that  $\mu$  has 0/1 coefficients, and that (14) and (13) hold. Notice that (14) is just a linear equation and proving (13) is proving the quadratic relation  $\mathbf{A}\mathbf{s}_1 + \mathbf{B}\mathbf{s}_2 + \mathbf{G}\mu\mathbf{s}_2 = \mathbf{u} \bmod q$ . All of these proofs fit into the appropriate functions in the protocol in Figure 10 and the full description of the group signature is given in Section 6.4.

The security of the scheme rests on the fact that creating a valid proof on a  $\mu$  that is not the user’s identity implies having a solution to (13) on a new identity, which is directly equivalent to breaking the ABB signature scheme [1, 38], which in turn implies breaking the Module-SIS problem. Prior to this work, proving tight bounds on the  $\ell_2$  norm of polynomial vectors with somewhat large coefficients was not very efficient, and so constructions of group signature schemes using this approach [16, 32] did not prove (13), but rather proved an approximate version of it as in (2) – i.e. they proved knowledge of  $\bar{\mathbf{s}}_1, \bar{\mathbf{s}}_2, c$  satisfying

$$[\mathbf{A} \mid \mathbf{B} + \mu\mathbf{G}] \cdot \begin{bmatrix} \bar{\mathbf{s}}_1 \\ \bar{\mathbf{s}}_2 \end{bmatrix} = c\mathbf{u} \bmod q, \quad (15)$$

where  $\|\bar{\mathbf{s}}_i\| \gg \|\mathbf{s}_i\|$ .

A consequence of being only able to prove the above is a vicious cycle of the larger norms and the presence of  $c$ , implying a larger extracted solution to the Module-SIS problem, which in turn requires a larger modulus, which also requires a larger lattice dimension. Furthermore, because these schemes relied on the verifiable encryption scheme of [31], they also did not prove (14), but rather an approximate version of it as in (2). The implication is that in order to decrypt, the Opener needed to guess the unknown  $c$ , which in expectation requires the same number of guesses as the adversary’s number of calls to the random oracle during the proof. Thus special care would be needed to instantiate the scheme in an environment that would not allow the adversary to be able to have too much time to try and forge a signature. We believe that efficiently eliminating this requirement in all lattice-based schemes requiring a verifiable encryption scheme is a notable improvement on the state of affairs.

	Public Key Size	Signature Size	Opening Time Independent of Adversary’s Forgery Time
[32]	96KB	203KB	×
This Work	48KB	90KB	✓

**Table 1:** Our group signature and that of [32].

We compare the instantiation of the group signature from this paper to the previously most efficient one from [32] in Table 1. We mention that there are also tree-based group signatures (e.g. [21, 11]) which have shorter outputs for small group sizes, but have the disadvantage that the signing time, verification time, and public key size are linear in the group size. The signature length of these schemes also grows slightly with the group size, and for groups having more than  $\approx 2^{21}$  members, our scheme has a comparable signature size (in addition to a much smaller public key and signing/verification times).

Part of the group signature includes a verifiable encryption scheme, in which the encryptor proves that the encryption is valid. When looked at separately, this scheme has a similar size to the one from [31], but with the noticeable advantage of not having a dependency between the decryption time and the adversary’s forgery time. We also give a comparison of the proof size for the basic system in (1) between our proof

<sup>11</sup> This paper presents a verifiable *decryption* scheme, but the proof size for a verifiable encryption scheme constructed in the same manner would be similar. At the very least, it needs to be as large as the proof of the basic equation in (1).

	Proof Size	Ciphertext Size	Proof Size	Decryption Time Independent of Forgery Time
[34]	33KB			
This Work	14KB			
		9KB	9KB	×
		4KB	33 - 44KB	✓
		1KB	19KB	✓

**Table 2:** The table on the left compares the difference in proof size of proving knowledge of short  $\vec{s}, \vec{e}$  satisfying  $A\vec{s} + \vec{e} = \vec{t} \pmod q$ , where  $A \in \mathbb{Z}_q^{1024 \times 1024}$  and  $q \approx 2^{32}$ , and  $\|(\vec{s}, \vec{e})\| \leq \sqrt{2048}$ . The protocol from [34] needs to make the additional restriction that all the coefficients in  $\vec{s}, \vec{e}$  are from  $\{-1, 0, 1\}$ . The table on the right compares our instantiation of a verifiable encryption scheme from this paper with [31] and [34].

system and the prior best one from [34] that followed the framework of [4] and [19]. The comparisons for the verifiable encryption scheme and the basic proof system are in table 2 and detailed descriptions of the proofs can be found in Appendices 6.2 and 6.3.

## 2 Preliminaries

### 2.1 Notation

Denote  $\mathbb{Z}_p$  to be the ring of integers modulo  $p$ . Let  $q = q_1, \dots, q_n$  be a product of  $n$  odd primes where  $q_1 < q_2 < \dots < q_n$ . Usually, we pick  $n = 1$  or  $n = 2$ . We write  $\vec{v} \in \mathbb{Z}_q^m$  to denote vectors over a ring  $\mathbb{Z}_q$ . Matrices over  $\mathbb{Z}_q$  will be written as regular capital letters  $R$ . By default, all vectors are column vectors. We write  $\vec{v} \parallel \vec{w}$  for a usual concatenation of  $\vec{v}$  and  $\vec{w}$  (which is still a column vector). For  $\vec{v}, \vec{w} \in \mathbb{Z}_q^k$ ,  $\vec{v} \circ \vec{w}$  is the usual component-wise multiplication. For simplicity, we denote  $\vec{u}^2 = \vec{u} \circ \vec{u}$ . We write  $x \leftarrow S$  when  $x \in S$  is sampled uniformly at random from the finite set  $S$  and similarly  $x \leftarrow D$  when  $x$  is sampled according to the distribution  $D$ . Let  $[n] := \{1, \dots, n\}$ .

For a power of two  $d$  and a positive integer  $p$ , denote  $\mathcal{R}$  and  $\mathcal{R}_p$  respectively to be the rings  $\mathbb{Z}[X]/(X^d + 1)$  and  $\mathbb{Z}_p[X]/(X^d + 1)$ . Lower-case letters denote elements in  $\mathcal{R}$  or  $\mathcal{R}_p$  and bold lower-case (resp. upper-case) letters represent column vectors (resp. matrices) with coefficients in  $\mathcal{R}$  or  $\mathcal{R}_p$ . For a polynomial  $f \in \mathcal{R}_p$ , denote  $\vec{f} \in \mathbb{Z}_q^d$  to be the coefficient vector of  $f$ . By default, we write its  $i$ -th coefficient as its corresponding regular font letter subscript  $i$ , e.g.  $f_{d/2} \in \mathbb{Z}_p$  is the coefficient corresponding to  $X^{d/2}$  of  $f \in \mathcal{R}_p$ . For the constant coefficient, however, we will denote  $\vec{f} := f_0 \in \mathbb{Z}_p$ . The ring  $\mathcal{R}$  has a group of automorphisms  $\text{Aut}(\mathcal{R})$  that is isomorphic to  $\mathbb{Z}_{2d}^\times$ . Let  $\sigma_i \in \text{Aut}(\mathcal{R}_q)$  be defined by  $\sigma_i(X) = X^i$ . For readability, we denote for an arbitrary vector  $\mathbf{m} \in \mathcal{R}^k$ :

$$\sigma_i(\mathbf{m}) := (\sigma_i(m_1), \dots, \sigma_i(m_k))$$

and similarly  $\sigma_i(\mathbf{R})$  for any matrix  $\mathbf{R}$ . When we write  $\langle \mathbf{u}, \mathbf{v} \rangle \in \mathbb{Z}$  for  $\mathbf{u}, \mathbf{v} \in \mathcal{R}^k$ , we mean the inner product of their corresponding coefficient vectors.

For an element  $w \in \mathbb{Z}_q$ , we write  $\|w\|_\infty$  to mean  $|w \pmod{\pm q}|$ . Define the  $\ell_\infty$  and  $\ell_p$  norms for  $w = w_0 + w_1X + \dots + w_{d-1}X^{d-1} \in \mathcal{R}$  as follows:

$$\|w\|_\infty = \max_j \|w_j\|_\infty, \quad \|w\|_p = \sqrt[p]{\|w_0\|_\infty^p + \dots + \|w_{d-1}\|_\infty^p}.$$

If  $\mathbf{w} = (w_1, \dots, w_m) \in \mathcal{R}^k$ , then

$$\|\mathbf{w}\|_\infty = \max_j \|w_j\|_\infty, \quad \|\mathbf{w}\|_p = \sqrt[p]{\|w_1\|_p^p + \dots + \|w_k\|_p^p}.$$

By default,  $\|\mathbf{w}\| := \|\mathbf{w}\|_2$ . Similarly, we define the norms for vectors over  $\mathbb{Z}_q$ . Denote  $S_\gamma = \{x \in \mathcal{R}_q : \|x\|_\infty \leq \gamma\}$ .

## 2.2 Probabilisty Distributions

We first define the discrete Gaussian distribution used for the rejection sampling.

**Definition 2.1.** *The discrete Gaussian distribution on  $\mathcal{R}^\ell$  centered around  $\mathbf{v} \in \mathcal{R}^\ell$  with standard deviation  $\mathfrak{s} > 0$  is given by*

$$D_{\mathbf{v}, \mathfrak{s}}^\ell(\mathbf{z}) = \frac{e^{-\|\mathbf{z}-\mathbf{v}\|^2/2\mathfrak{s}^2}}{\sum_{\mathbf{z}' \in \mathcal{R}^\ell} e^{-\|\mathbf{z}'\|^2/2\mathfrak{s}^2}}.$$

When it is centered around  $\mathbf{0} \in \mathcal{R}^\ell$  we write  $D_{\mathfrak{s}}^\ell = D_{\mathbf{0}, \mathfrak{s}}^\ell$ .

We will use the following tail bound, which follows from [5, Lemma 1.5(i)].

**Lemma 2.2.** *Let  $\mathbf{z} \leftarrow D_{\mathfrak{s}}^m$ . Then  $\Pr \left[ \|\mathbf{z}\| > t \cdot \mathfrak{s} \sqrt{md} \right] < \left( t e^{-\frac{t^2}{2}} \right)^{md}$ .*

Next, we recall the binomial distribution.

**Definition 2.3.** *The binomial distribution with a positive integer parameter  $\kappa$ , written as  $\text{Bin}_\kappa$  is the distribution  $\sum_{i=1}^\kappa (a_i - b_i)$ , where  $a_i, b_i \leftarrow \{0, 1\}$ . The variance of this distribution is  $\kappa/2$  and it holds that  $\text{Bin}_{\kappa_1} \pm \text{Bin}_{\kappa_2} = \text{Bin}_{\kappa_1 + \kappa_2}$ .*

## 2.3 Cyclotomic Rings

Suppose each  $(q_i)$  splits into  $l$  prime ideals of degree  $d/l$  in  $\mathcal{R}$ . This means  $X^d + 1 \equiv \varphi_0 \dots \varphi_{l-1} \pmod{q_i}$  with irreducible polynomials  $\varphi_j$  of degree  $d/l$  modulo  $q_i$ . We assume that  $\mathbb{Z}_q$  contains a primitive  $2l$ -th root of unity  $\zeta_i \in \mathbb{Z}_q$  but no elements whose order is a higher power of two, i.e.  $q_i - 1 \equiv 2l \pmod{4l}$ . Therefore, we have

$$X^d + 1 \equiv \prod_{j=0}^{l-1} \left( X^{\frac{d}{l}} - \zeta_i^{2j+1} \right) \pmod{q_i} \quad (16)$$

where  $\zeta_i^{2j+1}$  ( $j \in \mathbb{Z}_l$ ) ranges over all the  $l$  primitive  $2l$ -th roots of unity.

We recall the main result by Lyubashevsky and Seiler [37] which says that small polynomials over  $\mathcal{R}_{q_i}$  are invertible.

**Lemma 2.4 ([37]).** *Let  $p \equiv 2l + 1 \pmod{4l}$  be a prime. Then, any  $f \in \mathcal{R}_p$  which satisfies either  $0 < \|f\|_\infty < \frac{1}{\sqrt{l}} p^{1/l}$  or  $0 < \|f\| < p^{1/l}$  has an inverse in  $\mathcal{R}_p$ .*

Denote  $\mathcal{R}_q^\times$  to be the set of invertible polynomials in  $\mathcal{R}_q$ . Recall that a polynomial  $f$  is invertible in  $\mathcal{R}_q$  if and only if for each  $i \in [n]$ ,  $f \pmod{q_i}$  is invertible in  $\mathcal{R}_{q_i}$ . Hence, Lemma 2.4 says that if  $f \in \mathcal{R}_q$  satisfies  $0 < \|f\|_\infty < \frac{1}{\sqrt{l}} q_1^{1/l}$  or  $0 < \|f\| < q_1^{1/l}$  then  $f \in \mathcal{R}_q^\times$ .

The ring  $\mathcal{R}$  has a group of automorphisms  $\text{Aut}(\mathcal{R})$  that is isomorphic to  $\mathbb{Z}_{2d}^\times$ ,

$$i \mapsto \sigma_i : \mathbb{Z}_{2d}^\times \rightarrow \text{Aut}(\mathcal{R}),$$

where  $\sigma_i$  is defined by  $\sigma_i(X) = X^i$ . Consider  $\sigma_{-1} \in \text{Aut}(\mathcal{R}_q)$ . We define the following map  $\mathsf{T} : \mathbb{Z}^{kd} \times \mathbb{Z}^{kd} \rightarrow \mathcal{R}$  which given vectors  $\vec{a} = (a_0, \dots, a_{kd-1})$  and  $\vec{b} = (b_0, \dots, b_{kd-1})$ , it outputs:

$$\mathsf{T}(\vec{a}, \vec{b}) := \sum_{i=0}^{k-1} \sigma_{-1} \left( \sum_{j=0}^{d-1} a_{id+j} X^j \right) \cdot \left( \sum_{j=0}^{d-1} b_{id+j} X^j \right) \in \mathcal{R}. \quad (17)$$

As briefly described in the introduction and in more detail in Section 5, we will make use of the following simple property of  $\mathsf{T}$ .

**Lemma 2.5.** *Let  $\vec{a}, \vec{b} \in \mathbb{Z}^{kd}$  for  $k \geq 1$ . Then, the constant coefficient of  $\mathsf{T}(\vec{a}, \vec{b})$  is equal to  $\langle \vec{a}, \vec{b} \rangle$ .*

In Section 7 we show how to construct functions  $\mathsf{T}$  with the same property for different underlying rings than  $\mathbb{Z}[X]/(X^d + 1)$ .

## 2.4 Approximate Range Proofs

In some cases, we will not need to prove a tight bound on the norm of a vector, but it will be enough for us to prove that its coefficients are small. The application of this proof is in showing that the inner product of a vector is small enough that it is the same modulo  $q$  and over the integers. The intuition for obtaining such proofs is the observation that the inner product (modulo  $q$ ) of a random vector  $\vec{r} \leftarrow \text{Bin}_1^m$  with an arbitrary vector  $\vec{w} \in \mathbb{Z}_q^m$  is less than  $\frac{1}{2}\|\vec{w}\|$  with probability at most  $\frac{1}{2}$  [7]. The slightly more general lemma from [34] that we will be using is

**Lemma 2.6.** *Let  $\vec{w} \in \mathbb{Z}_q^m$  and  $\vec{y} \in \mathbb{Z}_q^k$ . Then*

$$\Pr_{R \leftarrow \text{Bin}_1^{k \times m}} \left[ \|R\vec{w} + \vec{y}\|_\infty < \frac{1}{2} \|\vec{w}\|_\infty \right] \leq 2^{-k}.$$

For a large  $m$ , the gap between the upper bound ( $m \cdot \|\vec{w}\|_\infty$ ) and the lower bound ( $\frac{1}{2} \cdot \|\vec{w}\|_\infty$ ) is a factor of  $m$ . One can probabilistically lower it to  $O(\sqrt{m})$ , but there is a way to get a constant-size gap by considering the  $\ell_2$ -norm. A well-known result of Johnson and Lindenstrauss says that any set of  $k$  points in  $m$ -dimensional Euclidean space can be embedded into a much smaller  $\ell$ -dimensional Euclidean space, where  $\ell = O(\log m)$  and independent of  $k$ , so that all pairwise distances are preserved within an arbitrarily small factor. In practical scenarios, such embeddings are simply random projections.

Recently, Gentry et al. [23] applied this result in the context of proving shortness of a committed vector  $\vec{w} \in \mathbb{Z}^m$ . Concretely, the idea is to choose a random rectangular matrix  $R \leftarrow \text{Bin}_2^{256 \times m}$  and prove that the projection  $\vec{v} = R\vec{w}$  with respect to  $R$  has small norm. When  $m$  not too small, substituting the continuous normal distribution by a binomial one (with the same variance) should heuristically result in very similar tail bounds. In [23], arguments regarding the moments of  $\text{Bin}_1$  and experimental results were used to support this heuristic. Using the fact that the distribution  $\|R \cdot 1^d\|$ , where entries of  $R$  are chosen from the normal distribution with mean 0 and variance  $\kappa/2$ , is the scaled  $\chi^2$  distribution with 256 degrees of freedom, i.e.  $\frac{\kappa}{2}m \cdot \chi^2[256]$ , we obtain the following (heuristic) generalization of [23][Corollary 3.2] (we only use this lemma for the case of  $\kappa = 1, 2$ ).

**Lemma 2.7.** *Under the heuristic substitution of  $\text{Bin}_\kappa$  with the normal distribution of variance  $\kappa/2$ , for any  $\vec{w} \in \mathbb{Z}^m$ ,*

1.  $\Pr_{R \leftarrow \text{Bin}_\kappa^{256 \times m}} [\|R\vec{w}\|^2 < \|\vec{w}\|^2 \cdot 13 \cdot \kappa] \approx \Pr_{y \leftarrow \chi^2[256]} [y < 26] \leq 2^{-256}$
2.  $\Pr_{R \leftarrow \text{Bin}_\kappa^{256 \times m}} [\|R\vec{w}\|^2 > \|\vec{w}\|^2 \cdot 337 \cdot \kappa] \approx \Pr_{y \leftarrow \chi^2[256]} [y > 674] \leq 2^{-128}.$

Gentry et al. construct a proof for the shortness of a long vector  $\vec{w} \in \mathbb{Z}_q^m$  as follows. They first commit to the random projection  $\vec{v} := R\vec{w} \in \mathbb{Z}_q^{256}$ , where  $R \leftarrow \text{Bin}_1^{256 \times m}$ , and prove that the norm of  $\vec{v}$  is small and that  $\vec{v}$  is a projection of  $\vec{w}$ . Then, [23][Corollary 3.3] says that if  $\|\vec{v}\| < b\sqrt{30}$ , where  $b \leq q/(45m)$ , then we must have  $\|\vec{w}\| \leq b$  (with an overwhelming probability). In our protocols, we will need a modified version of this result which says that for every vector  $\vec{y} \in \mathbb{Z}_q^{256}$ , if  $\|R\vec{w} + \vec{y}\|$  is small then we must have that  $\|\vec{w}\|$  is small. Even though we believe this generalisation is true for the constants described in [23][Corollary 3.3] (and a generalization for the analogous result in the  $\ell_\infty$  norm is true [35]), we don't know how to extend the proof to this setting. We thus provide a modified proof which results in slightly worse bounds.

**Lemma 2.8.** *Fix  $m, P \in \mathbb{N}$  and a bound  $b \leq P/41m$ , and let  $\vec{w} \in [\pm P/2]^m$  with  $\|\vec{w}\| \geq b$ , and let  $\vec{y}$  be an arbitrary vector in  $[\pm P/2]^m$ . Then*

$$\Pr_{R \leftarrow \text{Bin}_1^{256 \times m}} \left[ \|R\vec{w} + \vec{y} \bmod P\| < \frac{1}{2} b\sqrt{26} \right] < 2^{-128}.$$

*Proof.* We first prove an analogous result to [23][Corollary 3.3] with error  $2^{-256}$  rather than  $2^{-128}$ .

**Lemma 2.9.** Fix  $m, P \in \mathbb{N}$  and a bound  $b \leq P/41m$ , and let  $\vec{w} \in [\pm P/2]^m$  with  $\|\vec{w}\| \geq b$ . Then

$$\Pr_{R \leftarrow \text{Bin}_2^{256 \times m}} [\|R\vec{w} \bmod P\| < b\sqrt{26}] < 2^{-256}.$$

*Proof.* We have two cases:

- The first case is when  $\|\vec{w}\|_\infty \geq P/4m$ . Let  $i$  be an index of an entry in  $\vec{w}$  with magnitude at least  $P/4m$ , and consider any row of  $R$  (denoted  $\vec{r}$ ): After choosing all but the  $i$ 'th entry in  $\vec{r}$ , at most one of the three values  $\{0, \pm 1\}$  yields  $|\langle \vec{w}, \vec{r} \rangle \bmod P| < P/8m$ . Since the total probability of any two of those is at least  $1/2$  (i.e.  $\Pr[0] = 3/8$  and  $\Pr[\pm 1] = 1/4$ ), we have that the probability that all the rows of  $R$  yield entries smaller than  $P/8m$  is at most  $(1/2)^{256}$ . Since  $b \leq P/41m$  then  $P/8m > b\sqrt{26}$  and therefore

$$\Pr_{R \leftarrow \text{Bin}_2^{256 \times m}} [\|R\vec{w} \bmod P\| < b\sqrt{26}] \leq \Pr_R [\|R\vec{w} \bmod P\| < P/8m] \leq 2^{-256}.$$

- The second case is when  $\|\vec{w}\|_\infty < P/4m$ . Here with probability one we have  $R\vec{w} \in [\pm P/2]^{256}$ , so mod- $P$  reduction has no effect and the assertion follows directly from Lemma 2.7.  $\square$

We now use the above Lemma to prove Lemma 2.8. Suppose for contradiction that for some  $\vec{w}, \vec{y}$ ,

$$\Pr_{R \leftarrow \text{Bin}_1^{256 \times m}} \left[ \|R\vec{w} + \vec{y} \bmod P\| < \frac{1}{2}b\sqrt{26} \right] \geq 2^{-128}.$$

This implies that

$$\Pr_{R_1, R_2 \leftarrow \text{Bin}_1^{256 \times m}} \left[ \|R_1\vec{w} + \vec{y} \bmod P\| < \frac{1}{2}b\sqrt{26} \wedge \|R_2\vec{w} + \vec{y} \bmod P\| < \frac{1}{2}b\sqrt{26} \right] \geq 2^{-256}.$$

By the triangle inequality (which holds even modulo  $P$ ), we have

$$\Pr_{R_1, R_2 \leftarrow \text{Bin}_1^{256 \times m}} \left[ \|(R_1 - R_2)\vec{w} \bmod P\| < b\sqrt{26} \right] \geq 2^{-256}.$$

Since the distribution of  $R_1 - R_2$  is exactly  $\text{Bin}_2^{256 \times m}$ , the above implies that

$$\Pr_{R \leftarrow \text{Bin}_2^{256 \times m}} \left[ \|R\vec{w} \bmod P\| < b\sqrt{26} \right] \geq 2^{-256},$$

which is a contradiction with the statement of Lemma 2.9.  $\square$

## 2.5 Module-SIS and Module-LWE Problems

Security of the [6] commitment scheme used in our protocols relies on the well-known computational lattice problems, namely Module-LWE (MLWE) and Module-SIS (MSIS) [25]. Both problems are defined over  $\mathcal{R}_q$ .

**Definition 2.10** (MSIS $_{\kappa, m, B}$ ). Given  $\mathbf{A} \leftarrow \mathcal{R}_q^{\kappa \times m}$ , the Module-SIS problem with parameters  $\kappa, m > 0$  and  $0 < B < q$  asks to find  $\mathbf{z} \in \mathcal{R}_q^m$  such that  $\mathbf{A}\mathbf{z} = \mathbf{0}$  over  $\mathcal{R}_q$  and  $0 < \|\mathbf{z}\| \leq B$ . An algorithm  $\mathcal{A}$  is said to have advantage  $\epsilon$  in solving MSIS $_{\kappa, m, B}$  if

$$\Pr [0 < \|\mathbf{z}\| \leq B \wedge \mathbf{A}\mathbf{z} = \mathbf{0} \mid \mathbf{A} \leftarrow \mathcal{R}_q^{\kappa \times m}; \mathbf{z} \leftarrow \mathcal{A}(\mathbf{A})] \geq \epsilon.$$

**Definition 2.11** (MLWE $_{m, \lambda, \chi}$ ). The Module-LWE problem with parameters  $m, \lambda > 0$  and an error distribution  $\chi$  over  $\mathcal{R}$  asks the adversary  $\mathcal{A}$  to distinguish between the following two cases: 1)  $(\mathbf{A}, \mathbf{A}\mathbf{s} + \mathbf{e})$  for  $\mathbf{A} \leftarrow \mathcal{R}_q^{m \times \lambda}$ , a secret vector  $\mathbf{s} \leftarrow \chi^\lambda$  and error vector  $\mathbf{e} \leftarrow \chi^m$ , and 2)  $(\mathbf{A}, \mathbf{b}) \leftarrow \mathcal{R}_q^{m \times \lambda} \times \mathcal{R}_q^m$ . Then,  $\mathcal{A}$  is said to have advantage  $\epsilon$  in solving MLWE $_{m, \lambda, \chi}$  if

$$\begin{aligned} & \left| \Pr [b = 1 \mid \mathbf{A} \leftarrow \mathcal{R}_q^{m \times \lambda}; \mathbf{s} \leftarrow \chi^\lambda; \mathbf{e} \leftarrow \chi^m; b \leftarrow \mathcal{A}(\mathbf{A}, \mathbf{A}\mathbf{s} + \mathbf{e})] \right. \\ & \left. - \Pr [b = 1 \mid \mathbf{A} \leftarrow \mathcal{R}_q^{m \times \lambda}; \mathbf{b} \leftarrow \mathcal{R}_q^m; b \leftarrow \mathcal{A}(\mathbf{A}, \mathbf{b})] \right| \geq \epsilon. \end{aligned} \quad (18)$$

We also recall the (simplified) Extended Module-LWE problem [34].

**Definition 2.12 (Extended-MLWE $_{m,\lambda,\chi,C,s}$ ).** *The Extended Module-LWE problem with parameters  $m, \lambda > 0$ , probability distribution  $\chi$  over  $\mathcal{R}_q$ , challenge space  $\mathcal{C} \subseteq \mathcal{R}_q$  and the standard deviation  $\mathfrak{s}$  asks the adversary  $\mathcal{A}$  to distinguish between the following two cases:*

1.  $(\mathbf{B}, \mathbf{Br}, c, \mathbf{z}, \text{sign}(\langle \mathbf{z}, \mathbf{cr} \rangle))$  for  $\mathbf{B} \leftarrow \mathcal{R}_q^{m \times (m+\lambda)}$ , a secret vector  $\mathbf{r} \leftarrow \chi^{m+\lambda}$  and  $\mathbf{z} \leftarrow D_{\mathfrak{s}}^{(m+\lambda)d}$ ,  $c \leftarrow \mathcal{C}$
2.  $(\mathbf{B}, \mathbf{u}, c, \mathbf{z}, \text{sign}(\langle \mathbf{z}, \mathbf{cr} \rangle))$  for  $\mathbf{B} \leftarrow \mathcal{R}_q^{m \times (m+\lambda)}$ ,  $\mathbf{u} \leftarrow \mathcal{R}_q^m$ ,  $\mathbf{z} \leftarrow D_{\mathfrak{s}}^{(m+\lambda)d}$ ,  $c \leftarrow \mathcal{C}$ ,

where  $\text{sign}(a) = 1$  if  $a \geq 0$  and 0 otherwise. Then,  $\mathcal{A}$  is said to have advantage  $\epsilon$  in solving Extended-MLWE $_{m,\lambda,\chi,C,s}$  if

$$\left| \Pr \left[ b = 1 \mid \mathbf{B} \leftarrow \mathcal{R}_q^{m \times (m+\lambda)}; \mathbf{r} \leftarrow \chi^{m+\lambda}; \mathbf{z} \leftarrow D_{\mathfrak{s}}^{(m+\lambda)d}; c \leftarrow \mathcal{C}; b \leftarrow \mathcal{A}(\mathbf{B}, \mathbf{Br}, \mathbf{z}, c, s) \right] - \Pr \left[ b = 1 \mid \mathbf{B} \leftarrow \mathcal{R}_q^{m \times \lambda}; \mathbf{u} \leftarrow \mathcal{R}_q^m; \mathbf{z} \leftarrow D_{\mathfrak{s}}^{(m+\lambda)d}; c \leftarrow \mathcal{C}; b \leftarrow \mathcal{A}(\mathbf{B}, \mathbf{u}, \mathbf{z}, c, s) \right] \right| \geq \epsilon.$$

where  $s = \text{sign}(\langle \mathbf{z}, \mathbf{cr} \rangle)$ .

## 2.6 Rejection Sampling

In lattice-based zero-knowledge proofs, the prover will want to output a vector  $\mathbf{z}$  whose distribution should be independent of a secret message/randomness vector  $\mathbf{r}$ , so that  $\mathbf{z}$  cannot be used to gain any information on the prover's secret. During the protocol, the prover computes  $\mathbf{z} = \mathbf{y} + \mathbf{cr}$  where  $\mathbf{r}$  is either a secret vector or randomness used to commit to the prover's secret,  $c \leftarrow \mathcal{C}$  is a challenge polynomial, and  $\mathbf{y}$  is a "masking" vector. In order to remove the dependency of  $\mathbf{z}$  on  $\mathbf{r}$ , one applies *rejection sampling* [29].

**Lemma 2.13 (Rejection Sampling [29, 17, 34]).** *Let  $V \subseteq \mathcal{R}^\ell$  be a set of polynomials with norm at most  $T$  and  $\rho: V \rightarrow [0, 1]$  be a probability distribution. Fix the standard deviation  $\mathfrak{s} = \gamma T$ . Then, the following statements hold.*

1. Let  $M = \exp(12/\gamma + 1/(2\gamma^2))$ . Now, sample  $\mathbf{v} \leftarrow \rho$  and  $\mathbf{y} \leftarrow D_{\mathfrak{s}}^\ell$ , set  $\mathbf{z} = \mathbf{y} + \mathbf{v}$ , and run  $b \leftarrow \text{Rej}_1(\mathbf{z}, \mathbf{v}, \mathfrak{s})$  as defined in Fig. 1. Then, the probability that  $b = 0$  is at least  $(1 - 2^{-100})/M$  and the distribution of  $(\mathbf{v}, \mathbf{z})$ , conditioned on  $b = 0$ , is within statistical distance of  $2^{-100}$  of the product distribution  $\rho \times D_{\mathfrak{s}}^\ell$ .
2. Let  $M = \exp(1/(2\gamma^2))$ . Now, sample  $\mathbf{v} \leftarrow \rho$  and  $\mathbf{y} \leftarrow D_{\mathfrak{s}}^\ell$ , set  $\mathbf{z} = \mathbf{y} + \mathbf{v}$ , and run  $b \leftarrow \text{Rej}_2(\mathbf{z}, \mathbf{v}, \mathfrak{s})$  as defined in Fig. 1. Then, the probability that  $b = 0$  is at least  $1/(2M)$  and the distribution of  $(\mathbf{v}, \mathbf{z})$ , conditioned on  $b = 0$ , is identical to the distribution  $\mathcal{F}$  where  $\mathcal{F}$  is defined as follows: sample  $\mathbf{v} \leftarrow \rho$ ,  $\mathbf{z} \leftarrow D_{\mathfrak{s}}^{\ell d}$  conditioned on  $\langle \mathbf{v}, \mathbf{z} \rangle \geq 0$  and output  $(\mathbf{v}, \mathbf{z})$ .
3. Let  $M = \exp(1/(2\gamma^2))$ . Now, sample  $\mathbf{v} \leftarrow \rho, \beta \leftarrow \{0, 1\}$  and  $\mathbf{y} \leftarrow D_{\mathfrak{s}}^\ell$ , set  $\mathbf{z} = \mathbf{y} + (-1)^\beta \mathbf{v}$ , and run  $b \leftarrow \text{Rej}_0(\mathbf{z}, \mathbf{v}, \mathfrak{s})$  as defined in Fig. 2. Then, the probability that  $b = 0$  is at least  $1/M$  and the distribution of  $(\mathbf{v}, \mathbf{z})$ , conditioned on  $b = 0$ , is identical to the product distribution  $\rho \times D_{\mathfrak{s}}^\ell$ .

$\text{Rej}_1(\vec{z}, \vec{v}, \mathfrak{s})$ 01 $u \leftarrow [0, 1)$ 02 If $u > \frac{1}{M} \cdot \exp\left(\frac{-2\langle \vec{z}, \vec{v} \rangle + \ \vec{v}\ ^2}{2\mathfrak{s}^2}\right)$ 03 return 1 (i.e. reject) 04 Else 05 return 0 (i.e. accept)	$\text{Rej}_2(\vec{z}, \vec{v}, \mathfrak{s})$ 01 If $\langle \vec{z}, \vec{v} \rangle < 0$ 02 return 1 (i.e. reject) 03 $u \leftarrow [0, 1)$ 04 If $u > \frac{1}{M} \cdot \exp\left(\frac{-2\langle \vec{z}, \vec{v} \rangle + \ \vec{v}\ ^2}{2\mathfrak{s}^2}\right)$ 05 return 1 (i.e. reject) 06 Else 07 return 0 (i.e. accept)
--	---

**Fig. 1:** Two rejection sampling algorithms: the one used generally in previous works [29] (left) and the one proposed recently in [34] (right).

We recall how parameters  $\mathfrak{s}$  and  $M$  in the first statement Lemma 2.13 are selected. Concretely, the repetition rate  $M$  is chosen to be an upper-bound on:

$$\frac{D_{\mathfrak{s}}^{\ell}(\mathbf{z})}{D_{\mathbf{v},\mathfrak{s}}^{\ell}(\mathbf{z})} = \exp\left(\frac{-2\langle\mathbf{z},\mathbf{v}\rangle + \|\mathbf{v}\|^2}{2\mathfrak{s}^2}\right) \leq \exp\left(\frac{24\mathfrak{s}\|\mathbf{v}\| + \|\mathbf{v}\|^2}{2\mathfrak{s}^2}\right) = M. \quad (19)$$

For the inequality we used the which says that with probability at least  $1 - 2^{100}$  we have  $|\langle\mathbf{z},\mathbf{v}\rangle| < 12\mathfrak{s}\|\mathbf{v}\|$  for  $\mathbf{z} \leftarrow D_{\mathfrak{s}}^{\ell}$  [5, 29]. Hence, by setting  $\mathfrak{s} = 11\|\mathbf{v}\|$  we obtain  $M \approx 3$ .

Recently, Lyubashevsky et al. [34] proposed a modified rejection sampling algorithm (see  $\text{Rej}_2(\mathbf{z}, \mathbf{v}, \mathfrak{s})$  in Fig. 1) where it forces  $\mathbf{z}$  to satisfy  $\langle\mathbf{z}, \mathbf{v}\rangle \geq 0$ , otherwise it aborts. With this additional assumption, we can set  $M$  in the following way:

$$\exp\left(\frac{-2\langle\mathbf{z},\mathbf{v}\rangle + \|\mathbf{v}\|^2}{2\mathfrak{s}^2}\right) \leq \exp\left(\frac{\|\mathbf{v}\|^2}{2\mathfrak{s}^2}\right) = M. \quad (20)$$

Hence, for  $M \approx 3$  one would select  $\mathfrak{s} = 0.675 \cdot \|\mathbf{v}\|$ . Note that the probability for  $\mathbf{z} \leftarrow D_{\mathfrak{s}}^{\ell}$  that  $\langle\mathbf{z}, \mathbf{v}\rangle \geq 0$  is at least  $1/2$ . Hence, the expected number of rejections would be at most  $2M = 6$ . On the other hand, if one aims for  $M = 6$  repetitions using (19), then  $\mathfrak{s} = 6.74 \cdot \|\mathbf{v}\|$ . Thus, [34] manages to reduce the standard deviation by around a factor of 10. Further, we remark that this method is still not as efficient as using bimodal Gaussians [17], since even though the value  $M$  is calculated exactly as in (20), the expected number of rejections is at most  $M$  and not  $2M$ . We summarise the results from [17, 34] in the latter two statements of Lemma 2.13.

$\text{Rej}_0(\vec{z}, \vec{v}, \mathfrak{s})$ 01 $u \leftarrow [0, 1)$ 02 If $u > \frac{1}{M \exp\left(-\frac{\ \vec{v}\ ^2}{2\mathfrak{s}^2}\right) \cosh\left(\frac{\langle\vec{z}, \vec{v}\rangle}{\sigma^2}\right)}$ 03     return 1 (i.e. <i>reject</i> ) 04 Else 05     return 0 (i.e. <i>accept</i> )
--

**Fig. 2:** Bimodal rejection sampling [17].

Finally, we highlight that the procedure in the second statement of Lemma 2.13 reveals the sign of  $\langle\mathbf{z}, \mathbf{v}\rangle$ . This is still fine when working with “one-time commitments” [34] since we only leak one bit of information if  $\mathbf{v}$  is a randomness vector which is generated every execution. However, secure signature schemes cannot be produced using this method because each generation of a signature reveals some information about the secret key.

By using this technique, zero-knowledge property (or rather commit-and-prove simulatability as described in later sections) of our protocols relies on the (simplified) Extended-MLWE problem [34] where the adversary is given the additional one bit of information about the secret. We describe this problem in Section 2.5.

## 2.7 Challenge Space

In our applications, the set  $V \subseteq \mathcal{R}^{\ell}$  will consist of vectors of the form  $c\mathbf{r}$  where  $c \in \mathcal{R}_q$  is sampled from a challenge space  $\mathcal{C}$  and  $\mathbf{r} \in \mathcal{R}_q^{\ell}$  comes from a set of secret (either randomness or message) vectors. In order to set the standard deviation for rejection sampling, we need to bound the norm of such vectors. Here, we present a way to bound  $\|c\mathbf{v}\|$  which stems from the analysis in [34][Appendix C].

**Lemma 2.14.** *Let  $\mathbf{r} \in \mathcal{R}_q^{\ell}$  and  $c \in \mathcal{R}_q$ . Then  $\|c\mathbf{r}\| \leq \sqrt{\|\sigma_{-1}(c)\|} \|\mathbf{r}\|$ .*

*Proof.* Denote  $\mathbf{r} = (r_1, \dots, r_\ell) \in \mathcal{R}_q^\ell$  and  $C := \text{rot}(c)^T \text{rot}(c)$ . Then,

$$\begin{aligned} \|\mathbf{c}\mathbf{r}\|^2 &= \sum_{i=1}^{\ell} \|c r_i\|^2 = \sum_{i=1}^{\ell} \|\text{rot}(c) \vec{r}_i\|^2 = \sum_{i=1}^{\ell} \vec{r}_i^T \text{rot}(c)^T \text{rot}(c) \vec{r}_i \\ &= \sum_{i=1}^{\ell} \vec{r}_i^T C \vec{r}_i \\ &= \sum_{i=1}^{\ell} \sum_{\iota, j \in [d]} C_{\iota, j} r_{i, \iota} r_{i, j}. \end{aligned}$$

Now, we observe that  $C = \text{rot}(\sigma_{-1}(c)c)$  is skew-circulant and thus for each  $j \in [d]$  we have:

$$|C_{1, j}| = |C_{2, j+1}| = \dots = |C_{d-j+1, d}| = |C_{d-j+2, 1}| = \dots = |C_{d, j-1}|.$$

Thus, by the rearrangement inequality we get

$$\begin{aligned} \sum_{i=1}^{\ell} \sum_{\iota, j \in [d]} C_{\iota, j} r_{i, \iota} r_{i, j} &\leq \sum_{i=1}^{\ell} \sum_{j \in [d]} |C_{1, j}| (|r_{i, 1} r_{i, j}| + |r_{i, 2} r_{i, j+1}| + \dots + |r_{i, d} r_{i, j-1}|) \\ &\leq \sum_{i=1}^{\ell} \sum_{j \in [d]} |C_{1, j}| (r_{i, 1}^2 + \dots + r_{i, d}^2) \\ &\leq \sum_{i=1}^{\ell} \|\sigma_{-1}(c)c\|_1 \cdot \|\vec{r}_i\|^2 = \|\sigma_{-1}(c)c\|_1 \cdot \|\mathbf{r}\|^2. \end{aligned}$$

□

In order to apply this lemma, we set the challenge space  $\mathcal{C}$  as:

$$\mathcal{C} := \{c \in S_\kappa^\sigma : \sqrt{\|\sigma_{-1}(c)c\|_1} \leq \eta\} \quad (21)$$

where

$$S_\kappa^\sigma := \{c \in S_\kappa : \sigma(c) = c\}. \quad (22)$$

and the  $\sigma \in \text{Aut}(\mathcal{R}_q)$  will be specified in our protocols. Also, we denote  $\bar{\mathcal{C}} := \{c - c' : c, c' \in \mathcal{C} \text{ and } c \neq c'\}$  to be the set of differences of any two distinct elements in  $\mathcal{C}$ . In practice,  $\sigma \in \{\sigma_1, \sigma_{-1}\}$ . We will choose the constants  $\eta$  such that (experimentally) the probability for  $c \leftarrow S_\kappa^\sigma$  to satisfy  $\sqrt{\|\sigma_{-1}(c)c\|_1} \leq \eta$  is at least 99%. In our experiments, we observe that the bounds in Lemma 2.14 are about 4–6X larger than the actual norms  $\|\mathbf{c}\mathbf{r}\|$ .

For security of our protocols, we need  $\kappa < \frac{1}{2\sqrt{l}} q_1^{1/l}$  to ensure the invertibility property of the challenge space  $\mathcal{C}$ , i.e. the difference of any two distinct elements of  $\mathcal{C}$  is invertible over  $\mathcal{R}_q$  where  $l$  is the number of factors that  $X^d + 1$  splits into modulo  $q$ . Indeed, this property follows from the main result by Lyubashevsky and Seiler [37]. Secondly, to achieve negligible soundness error under the MSIS assumption, we will need  $|\mathcal{C}|$  to be exponentially large. In Table 3 we propose example parameters to instantiate the challenge space  $\mathcal{C}$  for different automorphisms  $\sigma$ . Finally, for implementation purposes, in order to sample from  $\mathcal{C}$ , we simply generate  $c \leftarrow S_\kappa^\sigma$  and check whether  $\sqrt{\|\sigma_{-1}(c)c\|_1} \leq \eta$ .

*Remark 2.15.* One could further reduce the value for  $\eta$  by letting the challenge be picked from  $S_\kappa^\sigma$  according to a different distribution than uniform as long as its min-entropy is at least 128. For example, if  $\sigma = \sigma_{-1}$  and we sample the first  $d/2$  polynomial coefficient from distribution  $\mathcal{D}$  such that  $\mathcal{D}(0) = \mathcal{D}(1) = \mathcal{D}(-1) = 1/4$  and  $\mathcal{D}(2) = \mathcal{D}(-2) = 1/8$ <sup>12</sup>, then  $\eta = 60$ . Hence, the improvement is not significant compared to the uniform case.

<sup>12</sup> The other ones are uniquely determined by the condition  $\sigma(c) = c$ .



$\sigma$	$d$	$l$	$\kappa$	$\eta$	$ S_\kappa^\sigma $	$ \mathcal{C} $
$\sigma_1$	128	8	1	37	$2^{202}$	$2^{201}$
$\sigma_{-1}$	128	4	2	72	$2^{148}$	$2^{147}$

**Fig. 3:** Example parameters to instantiate the challenge space  $\mathcal{C} := \{c \in S_\kappa : \sigma(c) = c \wedge \sqrt{\|\sigma_{-1}(c)\|_1} \leq \eta\}$  for a modulus  $q$  such that its smallest prime divisor  $q_1$  is greater than  $2^{20}$ .

**Setting the Standard Deviation.** By definition of the challenge space  $\mathcal{C}$  and Lemma 2.14, if we know that  $\|\mathbf{r}\| \leq \alpha$ , then we can set the standard deviation  $\mathfrak{s} := \gamma\eta\alpha$  where  $\gamma > 0$  defines the repetition rate  $M$ . On the other hand, if  $\|\mathbf{r}\|_\infty \leq \nu$ , e.g. because  $\mathbf{r} \leftarrow S_\nu^\ell$ , then we can set  $\mathfrak{s} := \gamma\nu\eta\sqrt{\ell n}$ .

### 3 The ABDLOP Commitment Scheme and Proofs of Linear Relations

In this section we formally present the ABDLOP commitment scheme together with ZKPoK of the committed messages. In the same protocol, we also include a proof of knowledge that the committed messages satisfy some arbitrary linear relations over  $\mathcal{R}_q$  (Figure 4). We then show how one can use this commitment scheme and proof of knowledge to prove knowledge of linear relations over  $\mathbb{Z}_q$  (Figure 5). This latter proof is best modeled as a commit-and-prove protocol because it will be creating some intermediate commitments under the same randomness, which cannot be simulated. In particular, what we prove is that the view, for all possible committed messages, is computationally indistinguishable from commitments to 0.

#### 3.1 The ABDLOP Commitment Scheme

Figure 4 presents the ABDLOP commitment scheme, which commits to messages  $\mathbf{s}_1$  and  $\mathbf{m}$ , using randomness  $\mathbf{s}_2$ , and then proves knowledge of these messages and that they satisfy the relation  $\mathbf{R}_1\mathbf{s}_1 + \mathbf{R}_m\mathbf{s}_m = \mathbf{u}$ . The challenge space  $\mathcal{C}$  is as in (21). The standard deviations  $\mathfrak{s}_1$  and  $\mathfrak{s}_2$  are set as in Section 2.6 so as to provide a balance between the running time of the algorithm (the lower the values, the higher the probability that the protocol will need to be repeated) and the security of the commitment scheme based on the hardness of the MSIS problem (the higher the values, the easier the problem becomes). Because the most common way in which our commitment scheme will be used involves committing to some values, proving that they satisfy some relations, and then never using the commitment again, we use a more efficient rejection sampling ( $\text{Rej}_2$  in Figure 1) from [34], which ends up leaking one bit of the secret, on the *randomness* part of the commitment (i.e.  $\mathbf{s}_2$ ). If one will not be throwing out this commitment, then one should use  $\text{Rej}_1$  for everything.

The hiding property of the commitment scheme follows from the MLWE problem when  $\mathbf{s}_2$  is chosen from some distribution such that  $\left(\begin{bmatrix} \mathbf{A}_2 \\ \mathbf{B} \end{bmatrix}, \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{B} \end{bmatrix} \cdot \mathbf{s}_2\right)$  is indistinguishable from uniform. The zero-knowledge property of the protocol follows from the standard argument from [29, 34] showing that  $\mathbf{z}_1, \mathbf{z}_2$  are distributed according to  $D_{\mathfrak{s}_1}^{m_1}$  and  $D_{\mathfrak{s}_2}^{m_2}$  (possibly with 1 bit of leakage for the latter) independent of  $\mathbf{s}_1$  and  $\mathbf{s}_2$ . The correctness of the protocol then follows due to the fact that  $m_i d$ -dimensional integer vectors sampled from a discrete Gaussian with standard deviation  $\mathfrak{s}_i$  has norm at most  $\mathfrak{s}_i \sqrt{2m_i d}$  with overwhelming probability [5].

The commitment opening needs to be defined to be whatever one can extract from the protocol. Since the protocol is an approximate proof of knowledge, it does not prove knowledge of  $\mathbf{s}_1, \mathbf{s}_2$  satisfying  $\mathbf{A}_1\mathbf{s}_1 + \mathbf{A}_2\mathbf{s}_2 = \mathbf{t}_A$ , but instead an approximate proof as in (2). Lemma 3.1 states that under the assumption that the Module-SIS problem is hard, the extracted values  $(\bar{\mathbf{s}}_1, \bar{\mathbf{s}}_2)$  are unique and they satisfy the desired linear equation  $\mathbf{R}_1\bar{\mathbf{s}}_1 + \mathbf{R}_m(\mathbf{t}_B - \mathbf{B}\bar{\mathbf{s}}_2) = \mathbf{u}$ , where  $\mathbf{m}$  is implicitly defined as  $\mathbf{t}_B - \mathbf{B}\bar{\mathbf{s}}_2$ . The last statement proved in the Lemma shows, as in [4], that not only are the extracted commitments  $\mathbf{s}_i$ , unique but also  $\mathbf{z}_i - c\bar{\mathbf{s}}_i$  is uniquely determined by the first two moves of the protocol. This is crucial to efficiently proving knowledge of polynomial products later in the paper.

As far as the communication complexity of the protocol, it is important to note that in the real protocol, one would not actually send  $\mathbf{w}$  and  $\mathbf{v}$ , but instead send their hash. Then one would verify the hash of the equalities. Therefore proving linear relations over  $\mathcal{R}_q$  is not any more costly, communication-wise, than just proving knowledge of the committed values. We don't write the hashes in our protocols because when they

eventually get converted to non-interactive ones using the Fiat-Shamir transform, the hashes will naturally enter the picture.

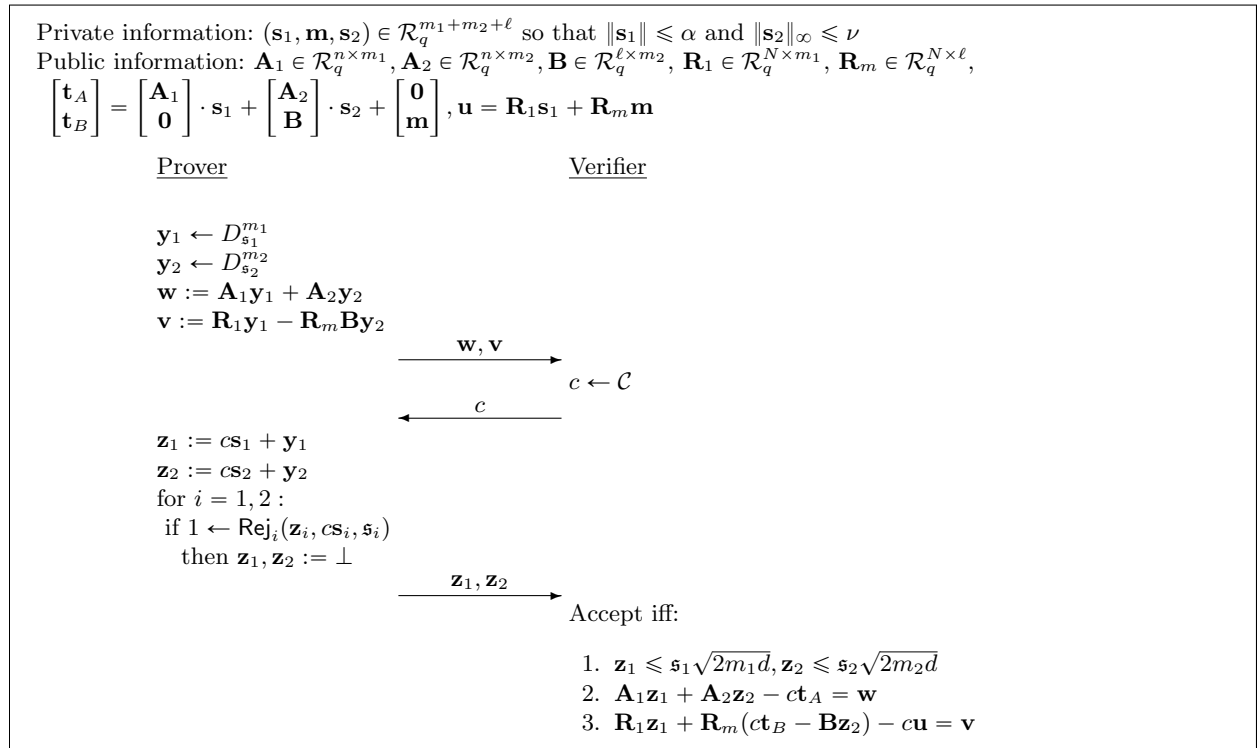
We will refer to the protocol in Figure 4 as  $\Pi_{\text{many}}^{(1)}((\mathbf{s}_2, \mathbf{s}_1, \mathbf{m}), (f_1, f_2, \dots, f_N))$ , where the  $f_i$  are linear functions mapping  $(\mathbf{s}_1, \mathbf{m})$  to  $\mathcal{R}_q$  such that  $f_i(\mathbf{s}_1, \mathbf{m}) = 0$ , represented by the rows of  $\mathbf{R}_1, \mathbf{R}_m$ , and  $\mathbf{u}$ .

**Lemma 3.1.** *The protocol in Figure 4 is a proof of knowledge of  $(\bar{\mathbf{s}}_1, \bar{\mathbf{s}}_2, \bar{c}) \in \mathcal{R}_q^{m_1} \times \mathcal{R}_q^{m_2} \times \bar{\mathcal{C}}$  satisfying*

1.  $\mathbf{A}_1 \bar{\mathbf{s}}_1 + \mathbf{A}_2 \bar{\mathbf{s}}_2 = \mathbf{t}_A$
2.  $\|\bar{\mathbf{s}}_i \bar{c}\| \leq 2\mathfrak{s}_i \sqrt{2m_i d}$  for  $i = 1, 2$
3.  $\mathbf{R}_1 \bar{\mathbf{s}}_1 + \mathbf{R}_m (\mathbf{t}_B - \mathbf{B} \bar{\mathbf{s}}_2) = \mathbf{u}$

Furthermore, under the assumption that  $\text{MSIS}_{n, m_1 + m_2, B}$  is hard for  $B = 8\eta \sqrt{\mathfrak{s}_1^2 2m_1 d + \mathfrak{s}_2^2 2m_2 d}$ ,

4. This  $(\bar{\mathbf{s}}_1, \bar{\mathbf{s}}_2)$  is unique
5. For any two valid transcripts  $(\mathbf{w}, \mathbf{v}, c, \mathbf{z}_1, \mathbf{z}_2)$  and  $(\mathbf{w}, \mathbf{v}, c', \mathbf{z}'_1, \mathbf{z}'_2)$ , it holds that  $\mathbf{z}_i - c\bar{\mathbf{s}}_i = \mathbf{z}'_i - c'\bar{\mathbf{s}}_i$ .



**Fig. 4:** Proof of knowledge  $\Pi_{\text{many}}^{(1)}((\mathbf{s}_2, \mathbf{s}_1, \mathbf{m}), (f_1, f_2, \dots, f_N))$  of  $(\mathbf{s}_1, \mathbf{s}_2, \bar{c}) \in \mathcal{R}_q^{m_1} \times \mathcal{R}_q^{m_2} \times \bar{\mathcal{C}}$  satisfying (i)  $\mathbf{A}_1 \mathbf{s}_1 + \mathbf{A}_2 \mathbf{s}_2 = \mathbf{t}_A, \mathbf{B} \mathbf{s}_2 + \mathbf{m} = \mathbf{t}_B$  (ii)  $\|\mathbf{s}_i \bar{c}\| \leq 2\mathfrak{s}_i \sqrt{2m_i d}$  for  $i = 1, 2$  and (iii)  $f_i(\mathbf{s}_1, \mathbf{m}) = 0$  for  $i \in [N]$  where each  $f_1, \dots, f_N : \mathcal{R}_q^{m_1 + \ell} \rightarrow \mathcal{R}_q$  is a linear function. The linear functions  $f_i$  are represented by the corresponding rows of matrices  $\mathbf{u}, \mathbf{R}_1, \mathbf{R}_m$  and prove  $\mathbf{u} = \mathbf{R}_1 \mathbf{s}_1 + \mathbf{R}_m \mathbf{m}$  where  $\mathbf{R}_1^{N \times m_1}, \mathbf{R}_m^{N \times \ell}, \mathbf{u} \in \mathcal{R}_q^N$  are public.

*Proof.* Let  $(\mathbf{w}, \mathbf{v}, c, \mathbf{z}_1, \mathbf{z}_2)$  and  $(\mathbf{w}, \mathbf{v}, c', \mathbf{z}'_1, \mathbf{z}'_2)$  be two accepting transcripts which are obtained via rewinding the prover who sends  $\mathbf{w}, \mathbf{v}$  in the first step. Because the transcripts are accepting, they satisfy the second verification equation, and by subtracting the two equalities, we obtain

$$\mathbf{A}_1 \bar{\mathbf{z}}_1 + \mathbf{A}_2 \bar{\mathbf{z}}_2 - \bar{c} \mathbf{t}_A = 0, \quad (23)$$

where  $\bar{\mathbf{z}}_i = \mathbf{z}_i - \mathbf{z}'_i$  and  $\bar{c} = c - c'$ . Dividing the above equation by  $\bar{c}$ , we obtain Lemma statement 1 where  $\bar{\mathbf{s}}_i = \bar{\mathbf{z}}_i / \bar{c}$ . Because the first verification checks that  $\mathbf{z}_i \leq \mathfrak{s}_i \sqrt{2m_i d}$ , we know that  $\bar{\mathbf{z}}_i \leq \mathfrak{s}_i \sqrt{2m_i d}$ , and so

Lemma statement 2 is satisfied. By subtracting the two equalities satisfying the third verification equation, we obtain

$$\mathbf{R}_1 \bar{\mathbf{z}}_1 + \mathbf{R}_m (\bar{c} \mathbf{t}_B - \mathbf{B} \bar{\mathbf{z}}_2) - \bar{c} \mathbf{u} = \mathbf{0}. \quad (24)$$

Dividing by  $\bar{c}$  and plugging in  $\bar{\mathbf{s}}_i = \bar{\mathbf{z}}_i / \bar{c}$ , we get Lemma statement 3.

Now suppose that the extractor extracts another triplet  $(\bar{\mathbf{s}}'_1, \bar{\mathbf{s}}'_2, \bar{c}')$  with  $(\bar{\mathbf{s}}_1, \bar{\mathbf{s}}_2) \neq (\bar{\mathbf{s}}'_1, \bar{\mathbf{s}}'_2)$ , which, as we already proved, must satisfy the first two statements of the lemma. Then we have

$$\mathbf{A}_1 \bar{\mathbf{s}}_1 + \mathbf{A}_2 \bar{\mathbf{s}}_2 = \mathbf{A}_1 \bar{\mathbf{s}}'_1 + \mathbf{A}_2 \bar{\mathbf{s}}'_2, \quad (25)$$

and multiplying the above by  $\bar{c} \bar{c}'$  yields

$$\mathbf{A}_1 (\bar{\mathbf{s}}_1 - \bar{\mathbf{s}}'_1) \bar{c} \bar{c}' + \mathbf{A}_2 (\bar{\mathbf{s}}_2 - \bar{\mathbf{s}}'_2) \bar{c} \bar{c}' = \mathbf{0}. \quad (26)$$

By Lemma condition 2, we know that  $\bar{\mathbf{s}}_i \bar{c}, \bar{\mathbf{s}}'_i \bar{c}' \leq 2\mathfrak{s}_i \sqrt{2m_i d}$ , and so the above can be rewritten as

$$\mathbf{A}_1 (\bar{\mathbf{z}}_1 \bar{c}' - \bar{\mathbf{z}}'_1 \bar{c}) + \mathbf{A}_2 (\bar{\mathbf{z}}_2 \bar{c}' - \bar{\mathbf{z}}'_2 \bar{c}) = \mathbf{0}, \quad (27)$$

where  $\bar{\mathbf{z}}_i, \bar{\mathbf{z}}'_i \leq 2\mathfrak{s}_i \sqrt{2m_i d}$ . By Lemma 2.14, multiplication by  $c \in \mathcal{C}$  increases the  $\ell_2$  norm by a factor of  $\eta$ , where  $\eta$  is defined in Figure 3. Thus multiplication by  $\bar{c} \in \bar{\mathcal{C}}$  increases the norm by a factor of  $2\eta$ , and thus  $\bar{\mathbf{z}}_i \bar{c}' - \bar{\mathbf{z}}'_i \bar{c} \leq 8\eta \mathfrak{s}_i \sqrt{2m_i d}$ . If  $\text{MSIS}_{n, m_1 + m_2, B}$  is hard for  $B = 8\eta \sqrt{\mathfrak{s}_1^2 2m_1 d + \mathfrak{s}_2^2 2m_2 d}$ , it implies that  $\bar{\mathbf{z}}_i \bar{c}' - \bar{\mathbf{z}}'_i \bar{c} = \mathbf{0}$ , which means that  $\bar{\mathbf{s}}_i = \bar{\mathbf{z}}_i / \bar{c} = \bar{\mathbf{z}}'_i / \bar{c}' = \bar{\mathbf{s}}'_i$ , and this proves Lemma statement 4.

To prove Lemma statement 5, suppose that  $\mathbf{z}_i - c \bar{\mathbf{s}}_i = \mathbf{z}'_i - c' \bar{\mathbf{s}}'_i + \mathbf{r}$  for some  $\mathbf{r}$ . Then, we can rewrite this as  $\bar{\mathbf{z}}_i / \bar{c} = \bar{\mathbf{s}}_i + \mathbf{r} / \bar{c}$ . Since we already proved that  $\bar{\mathbf{z}}_i / \bar{c} = \bar{\mathbf{s}}_i$ , and the  $\bar{\mathbf{s}}_i$  are unique, it means that  $\mathbf{r} = \mathbf{0}$ .  $\square$

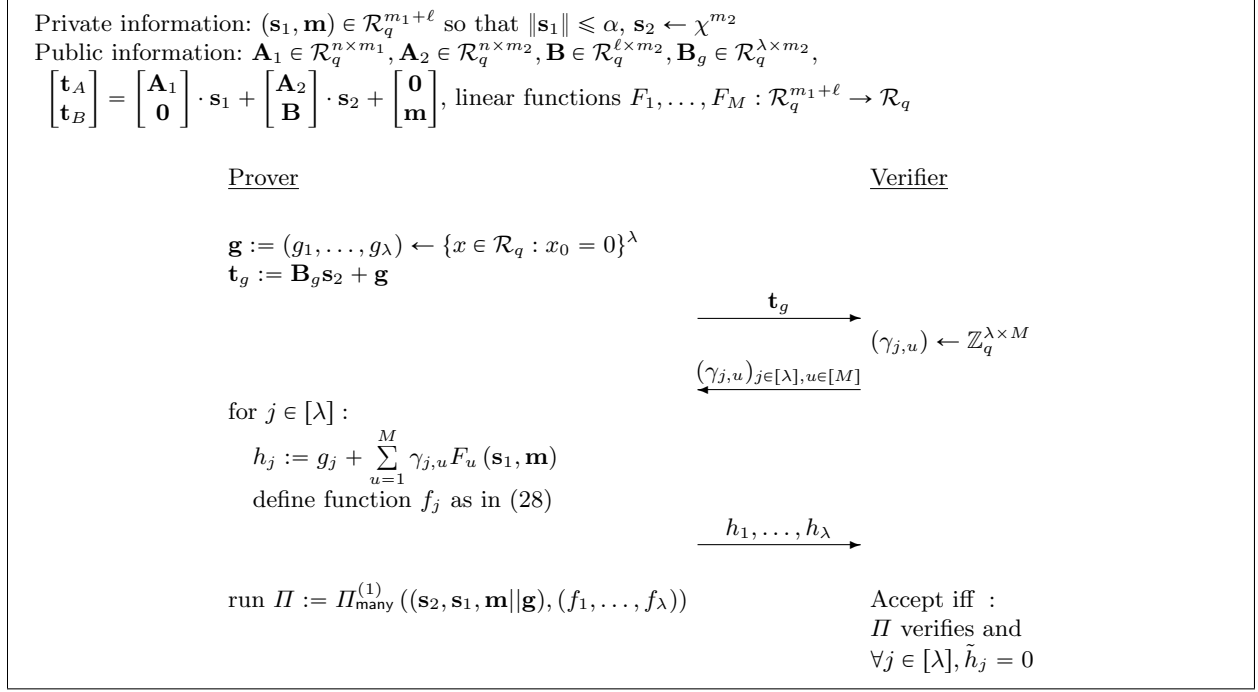
### 3.2 Linear Proofs over $\mathbb{Z}_q$

In this section we show how to transform the protocol from Figure 4 which proves that committed values satisfy a linear relation over  $\mathcal{R}_q$  into one that proves knowledge of the *constant coefficient* of a linear relation over  $\mathcal{R}_q$  (Figure 5). As shown in the introduction and Section 7, the inner product between two integer vectors appears in the constant coefficient of the polynomial product of two polynomials derived from these vectors. Thus proving knowledge that the constant coefficient of some linear function over  $\mathcal{R}_q$  is 0 is equivalent to proving knowledge that the output of a linear function over  $\mathbb{Z}_q$  is 0.

While it may see like proving knowledge of just the constant coefficient of a linear function over  $\mathcal{R}_q$  should not be much different than proving knowledge of the entire linear function as in Figure 4, the protocols do have some important differences. The main difference is that due to the need to mask all but the constant coefficient, we will need to create additional commitments during the proof. The most efficient way to do this is to append these commitments to the BDLOP part of the commitment scheme using the public randomness  $\mathbf{B}_g$  in Figure 5. The implication of needing to append committed values is that one can no longer reuse the commitment  $\mathbf{t}_A, \mathbf{t}_B$  since every run of the protocol essentially reveals more information about the randomness  $\mathbf{s}_2$ . Thus, instead of proving that the protocol is zero-knowledge, we show that the protocol is of a “commit-and-prove” type, where the security requirement is that the view of the commitment and the protocol output is computationally indistinguishable for all committed messages. All the other protocols in this paper also have this characteristic. This does not pose any problems for applications because the way we use a commitment scheme is in an auxiliary way to aid in proving that the value we care about satisfies some relations. Thus the commitment never needs to be reused.

The protocol begins by picking masking values  $g_i \in \mathcal{R}_q$ , which are uniformly random everywhere except in the constant coefficient, in which they are 0. These values are then appended to the commitment of  $\mathbf{m}$  as  $\mathbf{t}_g = \mathbf{B}_g \mathbf{s}_2 + \mathbf{g}$  and then sent to the verifier. The verifier picks  $\lambda$  random challenges for each of the  $M$  linear functions and the prover computes  $h_j = g_j + \sum_{u=1}^M \gamma_{j,u} F_u(\mathbf{s}_1, \mathbf{m})$  for each of the  $\lambda$  different  $j$ . Notice that the preceding are now linear functions

$$f_j(\mathbf{s}_1, \mathbf{m} | \mathbf{g}) := g_j + \sum_{u=1}^M \gamma_{j,u} F_u(\mathbf{s}_1, \mathbf{m}) - h_j \quad (28)$$



**Fig. 5:** Commit-and-prove protocol  $\Pi_{\text{eval}}^{(1)}((\mathbf{s}_2, \mathbf{s}_1, \mathbf{m}), (F_1, F_2, \dots, F_M))$  for messages  $(\mathbf{s}_1, \mathbf{m}) \in \mathcal{R}_q^{m_1+\ell}$ , randomness  $\mathbf{s}_2 \in \mathcal{R}_q^{m_2}$  and  $\bar{c} \in \bar{\mathcal{C}}$  which satisfy:  $\mathbf{A}_1 \mathbf{s}_1 + \mathbf{A}_2 \mathbf{s}_2 = \mathbf{t}_A$ ,  $\mathbf{B} \mathbf{s}_2 + \mathbf{m} = \mathbf{t}_B$  (ii)  $\|\mathbf{s}_i \bar{c}\| \leq 2s_i \sqrt{2m_i d}$  for  $i = 1, 2$  ( $\mathbf{s}_i$  are from Figure 4) and (iii) linear functions  $F_1, \dots, F_M : \mathcal{R}_q^{m_1+\ell} \rightarrow \mathcal{R}_q$  for which all the evaluations  $\tilde{F}_u(\mathbf{s}_1, \mathbf{m}) = 0$ . Here, we assume that the commitment  $(\mathbf{t}_A, \mathbf{t}_B)$  was generated honestly and already sent by the prover. In particular,  $\mathbf{s}_2 \leftarrow \chi^{m_2}$ .

over committed inputs  $\mathbf{s}_1, \mathbf{m} || \mathbf{g}$ . The prover completes the proof by sending the  $h_j$ , which completes the description of the functions, and begins the protocol in Figure 4 for proving that  $f_j(\mathbf{s}_1, \mathbf{m} || \mathbf{g}) = 0$ . The verifier accepts if the constant coefficient of  $h_j$  is 0 and the proof from Figure 4 is valid.

We now sketch the security and soundness properties of the protocol. This protocol is a warm-up for the full one in Figure 8 which proves knowledge of the constant coefficient of quadratic (rather than linear) functions over  $\mathcal{R}_q$ , and so we do not give a complete proof for it. To see that the view of the protocol is computationally indistinguishable for all messages  $\mathbf{s}_1, \mathbf{m}$ , we first observe that the full commitment that includes  $\mathbf{g}$  is indistinguishable from uniform based on (Extended)-Module-LWE as long as  $\left( \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{B} \\ \mathbf{B}_g \end{bmatrix}, \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{B} \\ \mathbf{B}_g \end{bmatrix} \cdot \mathbf{s}_2 \right)$  is indistinguishable from uniform when  $\mathbf{s}_2 \leftarrow \chi^{m_2}$ . To simulate the protocol, the simulator can simply pick  $\mathbf{t}_g$  uniformly at random and also choose  $h_1, \dots, h_\lambda$  at random (but having the first coefficient being 0). He can then simulate the protocol from figure 4 on the commitment  $(\mathbf{t}_A, \mathbf{t}_B, \mathbf{t}_g)$  and functions  $f_j$ . Thus the distribution is computationally indistinguishable from the correct one and is independent of the messages  $\mathbf{s}_1, \mathbf{m}$ .

To show that this protocol indeed proves that  $\tilde{F}_u(\mathbf{s}_1, \mathbf{m}) = 0$ , notice that the probability over the challenges  $\gamma_j, u$  that the equation  $h_j = g_j + \sum_{u=1}^M \gamma_{j,u} F_u(\mathbf{s}_1, \mathbf{m})$  is satisfied when  $\tilde{h}_j = 0$  and yet some  $\tilde{F}_u(\mathbf{s}_1, \mathbf{m}) \neq 0$  is at most  $1/q_1$ , where  $q_1$  is the smallest prime factor of  $q$ . The above holds because the values  $\mathbf{s}_1, \mathbf{m}$ , and  $\mathbf{g}$  were committed to prior to the verifier sending the challenges. The latter, as well as the fact

that the linear equations  $f_j$  are satisfied, is proved by the protocol  $\Pi_{\text{many}}^{(1)}((\mathbf{s}_2, \mathbf{s}_1, \mathbf{m} | \mathbf{g}), (f_1, \dots, f_\lambda))$ . The soundness error of the protocol is therefore  $q_1^{-\lambda}$ .

## 4 Proofs of Quadratic Relations

In this section we show how to prove various quadratic equations between committed messages using the ABDLOP commitment. More concretely, suppose we have message vectors  $\mathbf{s}_1 \in \mathcal{R}_q^{m_1}$  and  $\mathbf{m} \in \mathcal{R}_q^\ell$  such that  $\|\mathbf{s}_1\| \leq \alpha$ . Let  $\sigma \in \text{Aut}(\mathcal{R}_q)$  be a public automorphism over  $\mathcal{R}$  of degree  $k$  and for presentation purposes define:

$$(\sigma^i(\mathbf{x}))_{i \in [k]} := (\mathbf{x}, \sigma(\mathbf{x}), \dots, \sigma^{k-1}(\mathbf{x})) \in \mathcal{R}_q^{ka}$$

for arbitrary vector  $\mathbf{x} \in \mathcal{R}_q^a$ . Then, we consider the following statements:

- *Single quadratic equation with automorphisms.* For a public  $k(m_1 + \ell)$ -variate quadratic function  $f$  over  $\mathcal{R}_q$ ,

$$f((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]}) = 0.$$

- *Many quadratic equations with automorphisms.* For  $N$  public  $k(m_1 + \ell)$ -variate quadratic functions  $f_1, \dots, f_N$  over  $\mathcal{R}_q$ ,

$$f_j((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]}) = 0 \text{ for } j \in [N].$$

- *Many quadratic equations with automorphisms and a proof that polynomial evaluations have no constant coefficients.* For  $N + M$  public  $k(m_1 + \ell)$ -variate quadratic functions  $f_1, \dots, f_N$  and  $F_1, \dots, F_M$  over  $\mathcal{R}_q$ , the following hold:

- $f_j((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]}) = 0$  for  $j \in [N]$ ,
- let  $x_j := F_j((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]}) \in \mathcal{R}_q$  for  $j \in [M]$ . Then  $\tilde{x}_1 = \dots = \tilde{x}_M = 0$ .

*Remark 4.1.* Similarly as for [4], our techniques can be easily generalized to prove higher degree relations. Concretely, if we want to prove degree  $k$  equations, we end up committing to  $k - 1$  additional garbage terms. Throughout this paper (apart from Section 6.5), however, we will only consider quadratic relations.

### 4.1 Single Quadratic Equation with Automorphisms

Let  $(\mathbf{t}_A, \mathbf{t}_B)$  be the commitment to the message pair  $(\mathbf{s}_1, \mathbf{m})$  under randomness  $\mathbf{s}_2$ , i.e.

$$\begin{bmatrix} \mathbf{t}_A \\ \mathbf{t}_B \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{0} \end{bmatrix} \cdot \mathbf{s}_1 + \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{B} \end{bmatrix} \cdot \mathbf{s}_2 + \begin{bmatrix} \mathbf{0} \\ \mathbf{m} \end{bmatrix}.$$

Suppose the prover wants to prove knowledge of the message

$$\mathbf{s} = \begin{bmatrix} (\sigma^i(\mathbf{s}_1))_{i \in [k]} \\ (\sigma^i(\mathbf{m}))_{i \in [k]} \end{bmatrix} \in \mathcal{R}_q^{k(m_1 + \ell)}$$

such that  $f(\mathbf{s}) = 0$  where  $f$  is a  $k(m_1 + \ell)$ -variate quadratic function over  $\mathcal{R}_q$ . Note that each function  $f$  can be written explicitly as:

$$f(\mathbf{s}) = \mathbf{s}^T \mathbf{R}_2 \mathbf{s} + \mathbf{r}_1^T \mathbf{s} + r_0$$

where  $r_0 \in \mathcal{R}_q$ ,  $\mathbf{r}_1 \in \mathcal{R}_q^{k(m_1 + \ell)}$  and  $\mathbf{R}_2 \in \mathcal{R}_q^{k(m_1 + \ell) \times k(m_1 + \ell)}$ .

In order to prove this relation, let us consider the protocol for proving linear equations over  $\mathcal{R}_q$  in Fig. 4. In the last round, the honest prover sends the *masked openings*  $\mathbf{z}_i = \mathbf{c}\mathbf{s}_i + \mathbf{y}_i$  of  $\mathbf{s}_i$  for  $i = 1, 2$  where the challenge space  $\mathcal{C}$  is defined as in (21) with the  $\sigma$  automorphism. Even though this is not the case for  $\mathbf{m}$ , we can define the masked opening of  $\mathbf{m}$  as

$$\mathbf{z}_m := \mathbf{c}\mathbf{t}_B - \mathbf{B}\mathbf{z}_2 = \mathbf{c}\mathbf{m} - \mathbf{B}\mathbf{y}_2.$$

By construction,  $\mathbf{z}_m$  can be computed by the verifier.

Define the following vectors  $\mathbf{y}$  and  $\mathbf{z}$ :

$$\mathbf{y} := \begin{bmatrix} (\sigma^i(\mathbf{y}_1))_{i \in [k]} \\ -(\sigma^i(\mathbf{B}\mathbf{y}_2))_{i \in [k]} \end{bmatrix} \in \mathcal{R}_q^{k(m_1+\ell)} \quad (29)$$

and

$$\mathbf{z} := \begin{bmatrix} (\sigma^i(\mathbf{z}_1))_{i \in [k]} \\ (\sigma^i(\mathbf{z}_m))_{i \in [k]} \end{bmatrix} = c \begin{bmatrix} (\sigma^i(\mathbf{s}_1))_{i \in [k]} \\ (\sigma^i(\mathbf{m}))_{i \in [k]} \end{bmatrix} + \begin{bmatrix} (\sigma^i(\mathbf{y}_1))_{i \in [k]} \\ -(\sigma^i(\mathbf{B}\mathbf{y}_2))_{i \in [k]} \end{bmatrix} = c\mathbf{s} + \mathbf{y}. \quad (30)$$

Here we used the fact that for  $c \in \mathcal{C}$ ,  $\sigma(c) = c$ . Then, we have

$$\mathbf{z}^T \mathbf{R}_2 \mathbf{z} + c\mathbf{r}_1^T \mathbf{z} + c^2 r_0 = c^2 (\mathbf{s}^T \mathbf{R}_2 \mathbf{s} + \mathbf{r}_1^T \mathbf{s} + r_0) + cg_1 + g_0 \quad (31)$$

where polynomials  $g_1$  and  $g_0$  are defined as:

$$g_1 = \mathbf{s}^T \mathbf{R}_2 \mathbf{y} + \mathbf{y}^T \mathbf{R}_2 \mathbf{s} + \mathbf{r}_1^T \mathbf{y}, \quad g_0 = \mathbf{y}^T \mathbf{R}_2 \mathbf{y}.$$

Hence, we want to prove that the quadratic term in the expression  $\mathbf{z}^T \mathbf{R}_2 \mathbf{z} + c\mathbf{r}_1^T \mathbf{z} + c^2 r_0$  vanishes. This is done by first sending a commitment  $t$  to the polynomial  $g_1$ , i.e.  $t = \mathbf{b}^T \mathbf{s}_2 + g_1$  as well as  $v := g_0 + \mathbf{b}^T \mathbf{y}_2$  in the clear. Then, given  $t$  and the masked opening  $\mathbf{z}_2$  of  $\mathbf{s}_2$ , the verifier can compute  $f = ct - \mathbf{b}^T \mathbf{z}_2 = cg_1 - \mathbf{b}^T \mathbf{y}_2$ . Finally, it checks whether

$$\mathbf{z}^T \mathbf{R}_2 \mathbf{z} + c\mathbf{r}_1^T \mathbf{z} + c^2 r_0 - f \stackrel{?}{=} v$$

which is a simple transformation of (31) when  $\mathbf{s}^T \mathbf{R}_2 \mathbf{s} + \mathbf{r}_1^T \mathbf{s} + r_0 = 0$ .

We present the full protocol in Fig. 6 which follows the commit-and-prove paradigm [14, 34]. Namely, we assume the prover has already sent the commitments  $\mathbf{t}_A, \mathbf{t}_B$  to the verifier using fresh randomness  $\mathbf{s}_2 \leftarrow \chi^{m_2}$ . Prover starts by sampling masking vectors  $\mathbf{y}_1 \leftarrow D_{\mathbf{s}_1}^{m_1}, \mathbf{y}_2 \leftarrow D_{\mathbf{s}_2}^{m_2}$  and computing  $\mathbf{w} = \mathbf{A}_1 \mathbf{y}_1 + \mathbf{A}_2 \mathbf{y}_2$ . Then, it calculates  $g_1 = \mathbf{s}^T \mathbf{R}_2 \mathbf{y} + \mathbf{y}^T \mathbf{R}_2 \mathbf{s} + \mathbf{r}_1^T \mathbf{y}$ , where  $\mathbf{y}$  is defined in (29), and the commitment  $t = \mathbf{b}^T \mathbf{s}_2 + g_1$  to  $g_1$ . Finally, the prover sets  $v = \mathbf{y}^T \mathbf{R}_2 \mathbf{y} + \mathbf{b}^T \mathbf{y}_2$  and sends  $\mathbf{w}, t, v$  to the verifier.

Next, given a challenge  $c \leftarrow \mathcal{C}$ , the prover computes  $\mathbf{z}_i = c\mathbf{s}_i + \mathbf{y}_i$  for  $i = 1, 2$  and applies rejection sampling. If it does not abort, the prover outputs  $\mathbf{z}_1, \mathbf{z}_2$ .

Eventually, the verifier checks whether  $\mathbf{z}_1$  and  $\mathbf{z}_2$  have small norms,  $\mathbf{A}_1 \mathbf{z}_1 + \mathbf{A}_2 \mathbf{z}_2 = \mathbf{w} + c\mathbf{t}_A$  and  $\mathbf{z}^T \mathbf{R}_2 \mathbf{z} + c\mathbf{r}_1^T \mathbf{z} + c^2 r_0 - f = v$  where  $\mathbf{z}$  is defined in (30) and  $f$  is defined as  $f = ct - \mathbf{b}^T \mathbf{z}_2$ .

**Security Analysis.** We summarise security properties of the protocol in Fig. 6 below.

**Theorem 4.2.** *Consider the protocol in Fig. 6 and let  $\chi = S_\nu$ . Suppose  $\mathbf{s}_1 = \gamma_1 \alpha \eta$  and  $\mathbf{s}_2 = \gamma_2 \nu \eta \sqrt{m_2 d}$  for some  $\gamma_1, \gamma_2 > 0$  where  $\eta$  is chosen as in Section 2.7.*

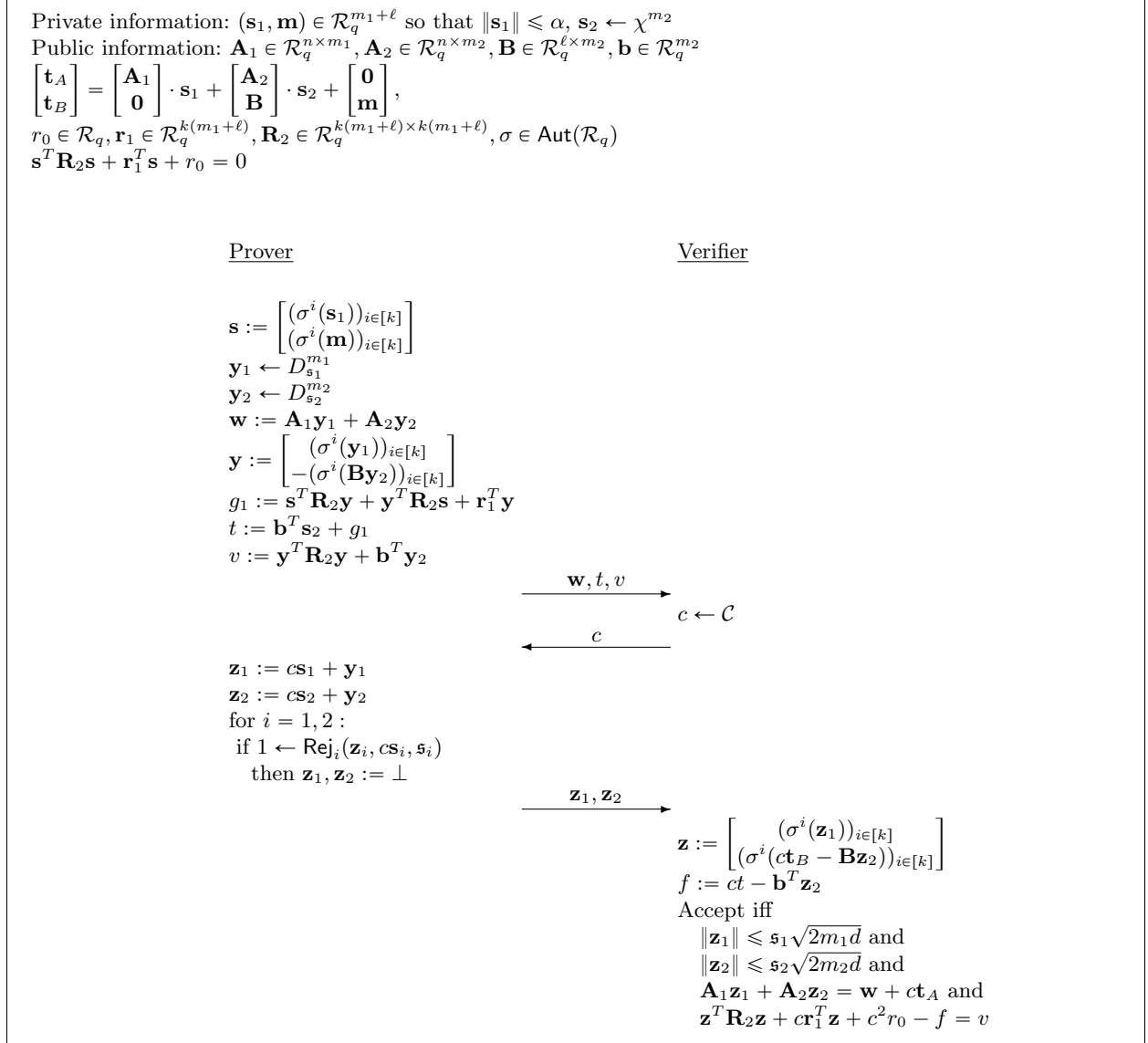
*For completeness, if  $m_1, m_2 \geq 512/d$  then the honest prover  $\mathcal{P}$  convinces the honest verifier  $\mathcal{V}$  with probability*

$$\approx 2 \exp \left( \frac{12}{\gamma_1} + \frac{1}{2\gamma_1^2} + \frac{1}{2\gamma_2^2} \right).$$

*For commit-and-prove simulatability, there exists a simulator  $\mathcal{S}$  that, without access to private information  $\mathbf{s}_1, \mathbf{m}$ , outputs a simulation of a commitment  $(\mathbf{t}_A, \mathbf{t}_B)$  along with a non-aborting transcript of the protocol between prover  $\mathcal{P}$  and verifier  $\mathcal{V}$  such that for every algorithm  $\mathcal{A}$  that has advantage  $\varepsilon$  in distinguishing the simulated commitment and transcript from the real commitment and transcript, whenever the prover does not abort, there is an algorithm  $\mathcal{A}'$  with the same running time that has advantage  $\varepsilon/2 - 2^{-100}$  in distinguishing the Extended-MLWE $_{n+\ell+1, m_2-n-\ell-1, \chi, \mathcal{C}, \mathbf{s}_2}$ .*

*For soundness, there is an extractor  $\mathcal{E}$  with the following properties. When given rewindable black-box access to a probabilistic prover  $\mathcal{P}^*$ , which convinces  $\mathcal{V}$  with probability  $\varepsilon \geq 2/|\mathcal{C}|$ ,  $\mathcal{E}$  either outputs  $(\bar{\mathbf{s}}_2, \bar{\mathbf{s}}_1, \bar{\mathbf{m}}) \in \mathcal{R}_q^{m_1+m_2+\ell}$  and  $\bar{c} \in \mathcal{R}_q^\times$  such that*

$$- \begin{bmatrix} \mathbf{t}_A \\ \mathbf{t}_B \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{0} \end{bmatrix} \cdot \bar{\mathbf{s}}_1 + \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{B} \end{bmatrix} \cdot \bar{\mathbf{s}}_2 + \begin{bmatrix} \mathbf{0} \\ \bar{\mathbf{m}} \end{bmatrix}$$



**Fig. 6:** Commit-and-prove protocol  $\Pi^{(2)}((\mathbf{s}_2, \mathbf{s}_1, \mathbf{m}), \sigma, f)$  for messages  $(\mathbf{s}_1, \mathbf{m}) \in \mathcal{R}_q^{m_1+\ell}$ , randomness  $\mathbf{s}_2 \in \mathcal{R}_q^{m_2}$  and  $\bar{c} \in \bar{\mathcal{C}}$  which satisfy:  $\mathbf{A}_1 \mathbf{s}_1 + \mathbf{A}_2 \mathbf{s}_2 = \mathbf{t}_A$ ,  $\mathbf{B} \mathbf{s}_2 + \mathbf{m} = \mathbf{t}_B$  (ii)  $\|\mathbf{s}_i \bar{c}\| \leq 2\mathbf{s}_i \sqrt{2m_i d}$  for  $i = 1, 2$  and (iii)  $f((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]}) = 0$  where function  $f: \mathcal{R}_q^{k(m_1+\ell)} \rightarrow \mathcal{R}_q$  is defined as  $f(\mathbf{x}) := \mathbf{x}^T \mathbf{R}_2 \mathbf{x} + \mathbf{r}_1^T \mathbf{x} + r_0$ . Here, we assume that the commitment  $(\mathbf{t}_A, \mathbf{t}_B)$  was generated honestly and already sent by the prover. In particular,  $\mathbf{s}_2 \leftarrow \chi^{m_2}$ .

- $f((\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]}, (\sigma^i(\bar{\mathbf{m}}))_{i \in [k]}) = 0$
- $\|\bar{c}\|_\infty \leq 2\kappa$
- $\|\bar{c}\bar{\mathbf{s}}_1\| \leq 2\mathfrak{s}_1\sqrt{2m_1d}$  and  $\|\bar{c}\bar{\mathbf{s}}_2\| \leq 2\mathfrak{s}_2\sqrt{2m_2d}$

or a  $\text{MSIS}_{n, m_1+m_2, B}$  solution for  $[\mathbf{A}_1 \ \mathbf{A}_2]$  in expected time at most  $3T/(\varepsilon - 2/|\mathcal{C}|)$  where running  $\mathcal{P}^*$  once is assumed to take at most  $T$  time and  $B := 8\eta\sqrt{\mathfrak{s}_1^2 2m_1d + \mathfrak{s}_2^2 2m_2d}$ .

*Proof.* We first focus on completeness. To begin with, we bound the norm of  $c\mathbf{s}_1$  and  $c\mathbf{s}_2$ . Note that by Lemma 2.14 and the definition of  $\mathcal{C}$  in (21):

$$\begin{aligned} \|c\mathbf{s}_1\| &\leq \sqrt{\|\sigma_{-1}(c)c\|\sqrt{d}}\|\mathbf{s}_1\| \leq \alpha\eta \\ \|c\mathbf{s}_2\| &\leq \sqrt{\|\sigma_{-1}(c)c\|_1 \cdot m_2d}\|\mathbf{s}_1\|_\infty \leq \nu\eta\sqrt{m_2d}. \end{aligned}$$

Then, by Lemma 2.13, the probability that  $\text{Rej}_1$  and  $\text{Rej}_2$  do not abort is at least

$$\exp\left(\frac{12}{\gamma_1} + \frac{1}{2\gamma_1^2}\right) \cdot 2 \exp\left(\frac{1}{2\gamma_2^2}\right).$$

Furthermore, by Lemma 2.2 for  $t = \sqrt{2}$  and our assumption that  $m_1, m_2 \geq 512/d$ , the probability that  $\|\mathbf{z}_1\| \leq \mathfrak{s}_1\sqrt{2m_1d}$  and  $\|\mathbf{z}_2\| \leq \mathfrak{s}_2\sqrt{2m_2d}$  is overwhelming. The other verification equations hold based on the discussion above.

*Commit-and-prove simulatability.* We can simulate the commitment and a non-aborting transcript between the honest prover and the honest verifier in the following way.

First, we define a hybrid simulator  $\mathcal{S}_0$  which still knows secret information  $\mathbf{s}_1, \mathbf{m}$ . Given a challenge  $c \leftarrow \mathcal{C}$ , it honestly generates the commitment  $(\mathbf{t}_A, \mathbf{t}_B, t)$  under randomness  $\mathbf{s}_2 \leftarrow \chi^{m_2}$ . Further, it samples fresh masked opening  $\mathbf{z}_1 \leftarrow D_{\mathfrak{s}_1}^{m_1d}$  and  $\mathbf{z}_2 \leftarrow D_{\mathfrak{s}_2}^d$  conditioned on  $\langle \mathbf{s}_2, \mathbf{z}_2 \rangle \geq 0$ . Finally, it sets  $\mathbf{w} := \mathbf{A}_1\mathbf{z}_1 + \mathbf{A}_2\mathbf{z}_2 - c\mathbf{t}_A$  and  $v := \mathbf{z}^T \mathbf{R}_2 \mathbf{z} + c\mathbf{r}_1^T \mathbf{z} + c^2 r_0 - ct + \mathbf{b}^T \mathbf{z}_2$ . Then, by Lemma 2.13, the distribution of the commitment and a transcript output by  $\mathcal{S}_0$  is statistically close to the one in the actual non-aborting protocol.

Next, we define the simulator  $\mathcal{S}_1$ , which still knows secret information  $\mathbf{s}_1, \mathbf{m}$ , as follows. It runs identically as  $\mathcal{S}_0$  but instead of generating the commitment  $(\mathbf{t}_A, \mathbf{t}_B, t)$  honestly, it samples  $\mathbf{u} \leftarrow \mathcal{R}_q^{n+\ell+1}$  and sets

$$\begin{bmatrix} \mathbf{t}_A \\ \mathbf{t}_B \\ t \end{bmatrix} = \mathbf{u} + \begin{bmatrix} \mathbf{A}_1 \mathbf{s}_1 \\ \mathbf{m} \\ g_1 \end{bmatrix}. \quad (32)$$

We claim that if there is a PPT adversary  $\mathcal{A}$  distinguishes between the outputs of  $\mathcal{S}_0$  and  $\mathcal{S}_1$  with probability  $\varepsilon$ , then there exists a PPT adversary  $\mathcal{B}$  which solves the Extended-MLWE $_{n+\ell+1, m_2-n-\ell-1, \chi, \mathcal{C}, \mathbf{s}_2}$  with probability at least  $\varepsilon/2$ . Indeed, we can define  $\mathcal{B}$  as follows. Given an Extended-MLWE tuple  $(\mathbf{C}, \mathbf{u}, \mathbf{z}_2, b)$ , where

$$\mathbf{C} := \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{B} \\ \mathbf{b}^T \end{bmatrix},$$

$\mathcal{B}$  sets  $(\mathbf{t}_A, \mathbf{t}_B, t)$  as in (32) and simulates the rest of the transcripts identically as  $\mathcal{S}_0$  and  $\mathcal{S}_1$ . Then, it outputs the commitment and the transcript to  $\mathcal{A}$ . Let us assume that  $b = 1$ . Note that if  $\mathbf{u} = \mathbf{C}\mathbf{s}_2$  then the output of  $\mathcal{B}$  comes from the distribution of  $\mathcal{S}_0$ . Similarly, if  $\mathbf{u}$  was uniformly random, then the output of  $\mathcal{B}$  comes from the distribution of  $\mathcal{S}_1$ . Hence, conditioned on  $b = 1$ ,  $\mathcal{B}$  solves the Extended-MLWE problem with probability at least  $\varepsilon$ . Since the probability of  $b = 1$  is at least  $1/2$ , the statement follows.

Finally, we can simply set  $\mathcal{S}$  (which does not use any secret information) to proceed identically as  $\mathcal{S}_1$  but instead of defining  $(\mathbf{t}_A, \mathbf{t}_B, t)$  as in (32), it directly samples  $(\mathbf{t}_A, \mathbf{t}_B, t) \leftarrow \mathcal{R}_q^{n+\ell+1}$ . Then, the output distributions of  $\mathcal{S}$  and  $\mathcal{S}_1$  are identical. Hence, the statement holds by the hybrid argument.



*Soundness.* We apply the strategy by Attema et al. [3]. Namely, let  $H \in \{0, 1\}^{R \times N}$  be a binary matrix where the  $R$  rows correspond to the prover's randomness and  $N$  columns correspond to verifier's randomness, i.e. different choices for the challenge  $c$ . For simplicity, we denote  $H(r, c)$  to be the entry corresponding to randomness  $r$  and challenge  $c \in \mathcal{C}$ . Clearly, an extractor can check values of each entry in  $H$  in time at most  $T$ .

We define the following extractor  $\mathcal{E}$ :

1.  $\mathcal{E}$  first samples fresh randomness  $r$  and challenge  $c^{(0)} \leftarrow \mathcal{C}$ . Then, it checks if  $H(r, c^{(0)}) = 1$ . If not,  $\mathcal{E}$  aborts.
2. Otherwise,  $\mathcal{E}$  samples along row  $r$  *without replacement* until it finds two  $c^{(1)}, c^{(2)}$  such that  $H(r, c^{(0)}) = H(r, c^{(1)}) = H(r, c^{(2)}) = 1$ <sup>13</sup>.

By [3, Remark 2], the expected time of  $\mathcal{E}$  is at most  $3T$  and  $\mathcal{E}$  extracts three valid transcripts

$$\text{tr}^{(i)} = (\mathbf{w}, t, v, c^{(i)}, \mathbf{z}_1^{(i)}, \mathbf{z}_2^{(i)}) \text{ for } i = 0, 1, 2$$

with probability at least  $\varepsilon - 2/|\mathcal{C}|$ .

First we focus on  $\text{tr}^{(0)}$  and  $\text{tr}^{(1)}$ . Define

$$\bar{c} := c^{(1)} - c^{(0)} \text{ and } \bar{\mathbf{s}}_i = \frac{\mathbf{z}_i^{(1)} - \mathbf{z}_i^{(0)}}{c^{(1)} - c^{(0)}} \text{ for } i = 1, 2.$$

By construction, we  $\|\bar{c}\|_\infty \leq 2\kappa$ ,  $\|\bar{\mathbf{s}}_1\| \leq 2\mathfrak{s}_1\sqrt{2m_1d}$  and  $\|\bar{\mathbf{s}}_2\| \leq 2\mathfrak{s}_2\sqrt{2m_2d}$ . Moreover, we have  $\mathbf{A}_1\bar{\mathbf{s}}_1 + \mathbf{A}_2\bar{\mathbf{s}}_2 = \mathbf{t}_A$ . Further, we define the extracted message vector  $\bar{\mathbf{m}} := \mathbf{t}_B - \mathbf{B}\bar{\mathbf{s}}_2$  and  $\bar{g}_1 := t - \mathbf{b}^T\bar{\mathbf{s}}_2$ . Then, we have

$$\begin{bmatrix} \mathbf{t}_A \\ \mathbf{t}_B \\ t \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{0} \\ 0 \end{bmatrix} \cdot \bar{\mathbf{s}}_1 + \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{B} \\ \mathbf{b}^T \end{bmatrix} \cdot \bar{\mathbf{s}}_2 + \begin{bmatrix} \mathbf{0} \\ \bar{\mathbf{m}} \\ \bar{g}_1 \end{bmatrix}.$$

Next, let  $\bar{\mathbf{y}}_i := \mathbf{z}_i^{(1)} - c^{(1)}\bar{\mathbf{s}}_i = \mathbf{z}_i^{(0)} - c^{(0)}\bar{\mathbf{s}}_i$  for  $i = 1, 2$ . Moreover, consider the third transcript  $\text{tr}^{(2)}$  and define  $\mathbf{y}_i^{(2)} := \mathbf{z}_i^{(2)} - c^{(2)}\bar{\mathbf{s}}_i$  for  $i = 1, 2$ . Using the identical argument as in the proof of Lemma 3.1, either  $(\bar{\mathbf{y}}_1, \bar{\mathbf{y}}_2) = (\mathbf{y}_1^{(2)}, \mathbf{y}_2^{(2)})$  or  $\mathcal{E}$  has found a  $\text{MSIS}_{n, m_1+m_2, B}$  solution for the matrix  $[\mathbf{A}_1 \ \mathbf{A}_2]$ . From now on, we assume the former case.

Finally, let us define the following vectors:

$$\bar{\mathbf{s}} := \begin{bmatrix} (\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]} \\ (\sigma^i(\bar{\mathbf{m}}))_{i \in [k]} \end{bmatrix} \text{ and } \bar{\mathbf{y}} := \begin{bmatrix} (\sigma^i(\bar{\mathbf{y}}_1))_{i \in [k]} \\ -(\sigma^i(\mathbf{B}\bar{\mathbf{y}}_2))_{i \in [k]} \end{bmatrix}.$$

Then, from the verification equations we have

$$\mathbf{z}^{(i)T} \mathbf{R}_2 \mathbf{z}^{(i)} + c^{(i)} \mathbf{r}_1^T \mathbf{z}^{(i)} + c^{(i)2} r_0 - \left( c^{(i)} t - \mathbf{b}^T \mathbf{z}_2^{(i)} \right) = v \text{ for } i = 0, 1, 2 \quad (33)$$

where

$$\mathbf{z}^{(i)} := \begin{bmatrix} (\sigma^i(\mathbf{z}_1^{(i)}))_{i \in [k]} \\ (\sigma^i(c^{(i)} \mathbf{t}_B - \mathbf{B} \mathbf{z}_2^{(i)}))_{i \in [k]} \end{bmatrix} = c^{(i)} \bar{\mathbf{s}} + \bar{\mathbf{y}}.$$

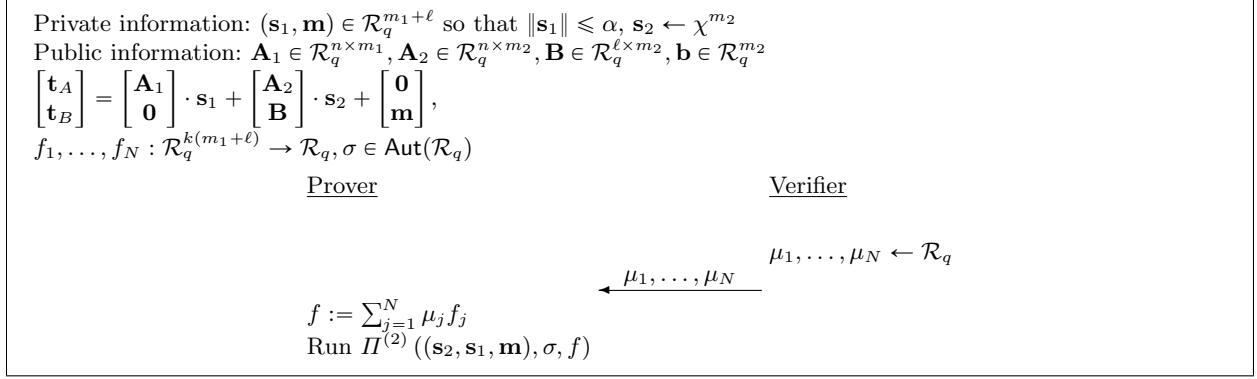
By expanding Equation 33, we obtain

$$c^{(i)2} (\bar{\mathbf{s}}^T \mathbf{R}_2 \bar{\mathbf{s}} + \mathbf{r}_1^T \bar{\mathbf{s}} + r_0) + c^{(i)} g'_1 + g'_0 = 0 \text{ for } i = 0, 1, 2$$

where

$$\begin{aligned} g'_1 &= \bar{\mathbf{s}}^T \mathbf{R}_2 \bar{\mathbf{y}} + \bar{\mathbf{y}}^T \mathbf{R}_2 \bar{\mathbf{s}} + \mathbf{r}_1^T \bar{\mathbf{y}} - \bar{g}_1 \\ g'_0 &= \bar{\mathbf{y}}^T \mathbf{R}_2 \bar{\mathbf{y}} + \mathbf{b}^T \bar{\mathbf{y}}_2 - v. \end{aligned}$$

<sup>13</sup> By construction,  $c^{(0)}, c^{(1)}, c^{(2)}$  are pairwise distinct.



**Fig. 7:** Commit-and-prove protocol  $\Pi_{\text{many}}^{(2)}((\mathbf{s}_2, \mathbf{s}_1, \mathbf{m}), \sigma, (f_1, f_2, \dots, f_N))$  for messages  $(\mathbf{s}_1, \mathbf{m}) \in \mathcal{R}_q^{m_1+\ell}$ , randomness  $\mathbf{s}_2 \in \mathcal{R}_q^{m_2}$  and  $\bar{c} \in \bar{\mathcal{C}}$  which satisfy:  $\mathbf{A}_1 \mathbf{s}_1 + \mathbf{A}_2 \mathbf{s}_2 = \mathbf{t}_A$ ,  $\mathbf{B} \mathbf{s}_2 + \mathbf{m} = \mathbf{t}_B$  (ii)  $\|\mathbf{s}_i \bar{c}\| \leq 2\mathfrak{s}_i \sqrt{2m_i d}$  for  $i = 1, 2$  (where  $\mathfrak{s}_i$  are used in Fig. 6) and (iii)  $f_j((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]}) = 0$  for  $j \in [N]$ . Vector  $\mathbf{b}$  is used in the sub-protocol  $\Pi^{(2)}$ .

Alternatively, we can write these three equations as follows:

$$\begin{bmatrix} 1 & c^{(0)} & c^{(0)^2} \\ 1 & c^{(1)} & c^{(1)^2} \\ 1 & c^{(2)} & c^{(2)^2} \end{bmatrix} \begin{bmatrix} g'_0 \\ g'_1 \\ \bar{\mathbf{s}}^T \mathbf{R}_2 \bar{\mathbf{s}} + \mathbf{r}_1^T \bar{\mathbf{s}} + r_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

Since the difference of each two challenges in  $\{c^{(0)}, c^{(1)}, c^{(2)}\}$  is invertible over  $\mathcal{R}_q$ , we must have that  $\bar{\mathbf{s}}^T \mathbf{R}_2 \bar{\mathbf{s}} + \mathbf{r}_1^T \bar{\mathbf{s}} + r_0 = 0$ . Hence, the statement holds.  $\square$

## 4.2 Many Quadratic Equations with Automorphisms

We consider a scenario when the prover wants to simultaneously prove  $N$  quadratic relations. Clearly, if one were to prove them separately using the approach from Section 4.1, one would end up committing to  $N$  garbage polynomials  $g$ . Here, we circumvent this issue by linear-combining the  $N$  equations into one quadratic equation and prove it using the protocol in Fig. 6. This results in committing to only one garbage polynomials at the cost of reducing the soundness error by a negligible additive factor.

More precisely, suppose that we want to prove for  $N$  public  $k(m_1 + \ell)$ -variate quadratic functions  $f_1, \dots, f_N$  over  $\mathcal{R}_q$  that

$$f_j((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]}) = 0 \text{ for } i \in [N]. \quad (34)$$

We let the verifier begin by sending challenges  $\mu_1, \dots, \mu_N \leftarrow \mathcal{R}_q$ . Then, we define a single quadratic function

$$f := \sum_{j=1}^N \mu_j f_j$$

and prove that

$$f((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]}) = 0 \quad (35)$$

using the protocol from Fig. 6. Now, we observe that if one of the conditions in (34) does not hold, then Equation 35 is satisfied with probability at most  $q_1^{-d/l}$  where  $l$  is the number of factors that  $X^d + 1$  splits into modulo  $q$ .

The protocol is provided in Fig. 7. We skip the security analysis since it will be implicitly included in the more general case in Theorem 4.3.

### 4.3 Polynomial Evaluations with Vanishing Constant Coefficients

Suppose we want to prove simultaneously  $N$  quadratic relations (i.e. (34)) and *additionally* prove that for quadratic  $k(m_1 + \ell)$ -variate polynomials  $F_1, \dots, F_M$ , evaluations  $F_j((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]})$  have the constant coefficient equal to zero. Concretely, if we denote

$$x_j := F_j((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]}) \in \mathcal{R}_q$$

then  $\tilde{x} = 0$  for  $j \in [M]$ .

For simplicity we first present an approach with soundness error  $1/q_1$ . We apply the strategy from [19] and first commit to a random masking polynomial  $g \leftarrow \{x \in \mathcal{R}_q : \tilde{x} = 0\}$ . Then, given random challenges  $\gamma_1, \dots, \gamma_M \leftarrow \mathbb{Z}_q$ , we send

$$h := g + \sum_{j=1}^M \gamma_j F_j((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]}) \quad (36)$$

to the verifier. Then, it simply checks whether the constant coefficient of  $h$  is indeed equal to zero. What is left to prove is that  $h$  is well-formed, i.e. (36) holds. This is done by defining the quadratic function  $f_{N+1} : \mathcal{R}_q^{k(m_1 + \ell + 1)} \rightarrow \mathcal{R}_q$  as follows.

Let  $\mathbf{x}_1 \in \mathcal{R}_q^{km_1}$ ,  $\mathbf{x}_2 = (\mathbf{x}_{2,1}, \dots, \mathbf{x}_{2,k}) \in \mathcal{R}_q^{k(\ell+1)}$  and denote

$$\mathbf{x}_{2,j} := \mathbf{x}_{2,j}^{(m)} \parallel x_{2,j}^{(g)} \in \mathcal{R}_q^{\ell+1} \text{ for } j \in [k], \quad \mathbf{x}_2^{(m)} := (\mathbf{x}_{2,1}^{(m)}, \dots, \mathbf{x}_{2,k}^{(m)}).$$

Then,

$$f_{N+1}(\mathbf{x}_1, \mathbf{x}_2) := x_{2,1}^{(g)} + \sum_{j=1}^M \gamma_j F_j(\mathbf{x}_1, \mathbf{x}_2^{(m)}) - h.$$

By construction, if  $(\mathbf{x}_1, \mathbf{x}_2) = (\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m} \parallel g))_{i \in [k]}$  then

$$\mathbf{x}_1 = \sigma^i(\mathbf{s}_1)_{i \in [k]}, \quad \mathbf{x}_2^{(m)} = \sigma^i(\mathbf{m})_{i \in [k]} \quad \text{and} \quad x_{2,1}^{(g)} = g.$$

Moreover, (36) holds if and only if

$$f_{N+1}((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m} \parallel g))_{i \in [k]}) = 0.$$

Recall that we also want to prove (34). We can define analogous polynomials  $f_1, \dots, f_N : \mathcal{R}_q^{k(m_1 + \ell + 1)} \rightarrow \mathcal{R}_q$  as:

$$f_j(\mathbf{x}_1, \mathbf{x}_2) := f_j(\mathbf{x}_1, \mathbf{x}_2^{(m)}).$$

Hence, we simply want to prove that for every  $j = 1, 2, \dots, N + 1$ :

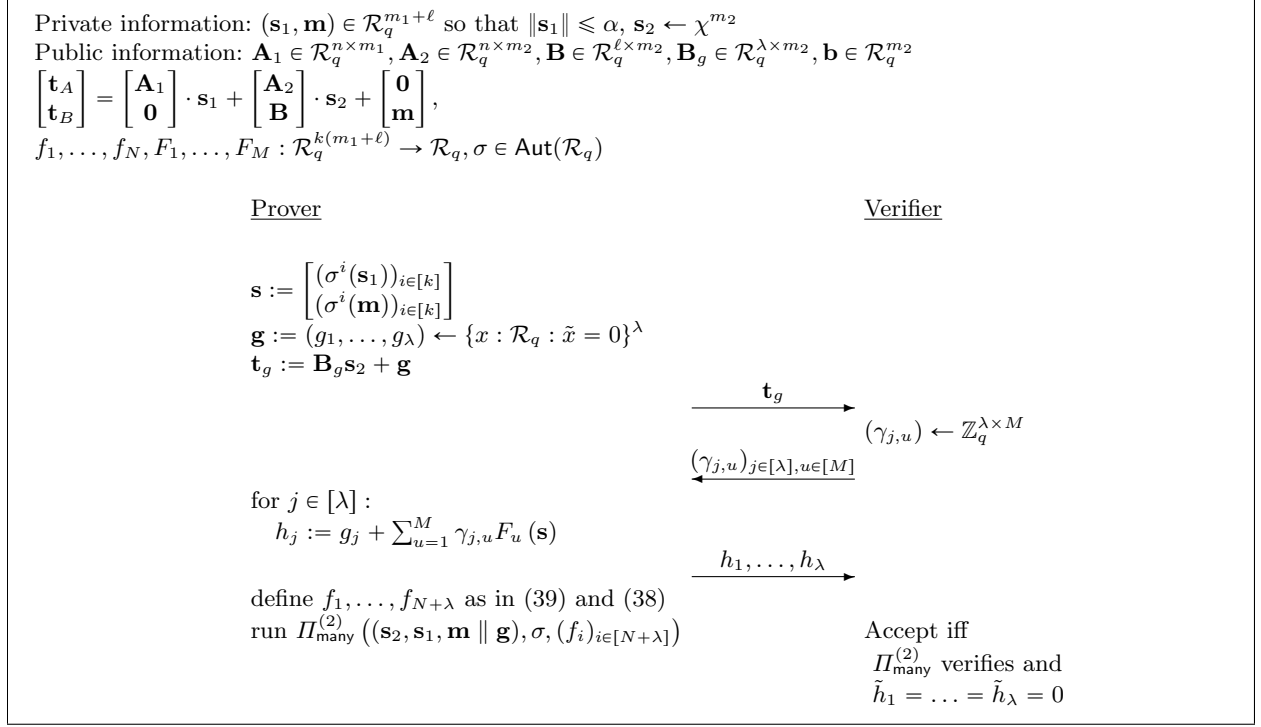
$$f_j((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m} \parallel g))_{i \in [k]}) = 0.$$

Finally, this can then be directly done using the protocol

$$\Pi_{\text{many}}^{(2)}((\mathbf{s}_2, \mathbf{s}_1, \mathbf{m}, g), \sigma, (f_1, f_2, \dots, f_{N+1}))$$

in Fig. 7.

We provide intuition for the soundness argument. Assume that the verifier is convinced that  $h$  is of the correct form (36) and  $\tilde{h} = 0$ . Also, note that a cheating prover committed to  $g$  before seeing the challenges  $\gamma_1, \dots, \gamma_M$ . Hence, if for some  $j \in [M]$ , the constant coefficient of  $F_j((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]})$  is non-zero, then the cheating prover has probability at most  $1/q_1$  of guessing the constant coefficient of  $\sum_{j=1}^M \gamma_j F_j((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]})$ .



**Fig. 8:** Commit-and-prove protocol  $\Pi_{\text{eval}}^{(2)}((\mathbf{s}_2, \mathbf{s}_1, \mathbf{m}), \sigma, (f_1, \dots, f_N), (F_1, \dots, F_M))$  for messages  $(\mathbf{s}_1, \mathbf{m}) \in \mathcal{R}_q^{m_1+\ell}$ , randomness  $\mathbf{s}_2 \in \mathcal{R}_q^{m_2}$  and  $\bar{c} \in \bar{\mathcal{C}}$  which satisfy:  $\mathbf{A}_1 \mathbf{s}_1 + \mathbf{A}_2 \mathbf{s}_2 = \mathbf{t}_A$ ,  $\mathbf{B} \mathbf{s}_2 + \mathbf{m} = \mathbf{t}_B$  (ii)  $\|\mathbf{s}_i \bar{c}\| \leq 2\mathbf{s}_i \sqrt{2m_i d}$  for  $i = 1, 2$ , (iii)  $f_j((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]}) = 0$  for  $j \in [N]$  (where  $\mathbf{s}_i$  are used in Fig. 6) and (iv) all the evaluations  $F_u((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]})$ , where  $u \in [M]$ , have constant coefficients equal to zero. Vector  $\mathbf{b}$  is used in the sub-protocol  $\Pi_{\text{many}}^{(2)}$ .

**Boosting Soundness.** We exponentially decrease the soundness error by parallel repetition. Namely, in order to obtain  $q_1^{-\lambda}$  soundness error, we commit to  $\lambda$  random masking polynomials  $\mathbf{g} = (g_1, \dots, g_\lambda) \leftarrow \{x : \mathcal{R}_q : \tilde{x} = 0\}^\lambda$  as follows:

$$\mathbf{t}_g := \mathbf{B}_g \mathbf{s}_2 + \mathbf{g}.$$

Then, we send  $\mathbf{t}_g$  to the verifier which in return outputs the challenge matrix  $(\gamma_{i,j})_{i \in [\lambda], j \in [M]} \leftarrow \mathbb{Z}_q^{\lambda \times M}$ . Then, we compute the vector  $\mathbf{h} = (h_1, \dots, h_\lambda)$  as follows:

$$\begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_\lambda \end{bmatrix} = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_\lambda \end{bmatrix} + \begin{bmatrix} \gamma_{1,1} & \gamma_{1,2} & \cdots & \gamma_{1,M} \\ \vdots & \vdots & \cdots & \vdots \\ \gamma_{\lambda,1} & \gamma_{\lambda,2} & \cdots & \gamma_{\lambda,M} \end{bmatrix} \begin{bmatrix} F_1((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]}) \\ F_2((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]}) \\ \vdots \\ F_M((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]}) \end{bmatrix} \quad (37)$$

and send it to the verifier. It directly checks if all polynomials  $h_1, \dots, h_\lambda \in \mathcal{R}_q$  have constant coefficients equal to zero.

As before, we still need to prove that vector  $\mathbf{h}$  was constructed correctly. We reduce this problem to proving quadratic relations. Namely, we define polynomials  $f_{N+1}, \dots, f_{N+M} : \mathcal{R}_q^{k(m_1+\ell+\lambda)} \rightarrow \mathcal{R}_q$  as follows.

Let  $\mathbf{x}_1 \in \mathcal{R}_q^{km_1}$ ,  $\mathbf{x}_2 = (\mathbf{x}_{2,1}, \dots, \mathbf{x}_{2,k}) \in \mathcal{R}_q^{k(\ell+\lambda)}$  and denote

$$\mathbf{x}_{2,j} := \left( \mathbf{x}_{2,j}^{(m)}, \mathbf{x}_{2,j}^{(g)} \right) \in \mathcal{R}_q^{\ell+\lambda} \text{ for } j \in [k],$$

$$\mathbf{x}_2^{(m)} := \left( \mathbf{x}_{2,1}^{(m)}, \dots, \mathbf{x}_{2,k}^{(m)} \right), \quad \mathbf{x}_{2,1}^{(g)} := \left( x_{2,1,1}^{(g)}, \dots, x_{2,1,\lambda}^{(g)} \right).$$

Then,

$$f_{N+j}(\mathbf{x}_1, \mathbf{x}_2) := x_{2,1,j}^{(g)} + \sum_{u=1}^M \gamma_{j,u} F_u(\mathbf{x}_1, \mathbf{x}_2^{(m)}) - h_j \text{ for } j \in [\lambda]. \quad (38)$$

By construction, if  $(\mathbf{x}_1, \mathbf{x}_2) = (\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m} \parallel \mathbf{g}))_{i \in [k]}$  then

$$\mathbf{x}_1 = \sigma^i(\mathbf{s}_1)_{i \in [k]}, \quad \mathbf{x}_2^{(m)} = (\sigma^i(\mathbf{m}))_{i \in [k]} \quad \text{and} \quad x_{2,1,j}^{(g)} = g_j.$$

Furthermore, Equation (37) is true if and only if for all  $j \in [\lambda]$  we have:

$$f_{N+j}((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m} \parallel \mathbf{g}))_{i \in [k]}) = 0.$$

Since we also need to prove (34), for convenience we define polynomials  $f_1, \dots, f_N : \mathcal{R}_q^{k(m_1 + \ell + \lambda)} \rightarrow \mathcal{R}_q$  as:

$$f_j(\mathbf{x}_1, \mathbf{x}_2) := f_j(\mathbf{x}_1, \mathbf{x}_2^{(m)}). \quad (39)$$

Finally, we simply run  $\Pi_{\text{quad-many}}((\mathbf{s}_2, \mathbf{s}_1, \mathbf{m}, \mathbf{g}), \sigma, (f_j)_{j \in [N+\lambda]})$  from Fig. 7. We summarise the protocol in Fig. 8 and provide commitment and proof size analysis in Section 6.1.

Note that with this approach we need to commit to additional  $\lambda$  garbage polynomials.

**Security Analysis.** We present the security properties of the protocol in Fig. 8 below.

**Theorem 4.3.** *Consider the protocol in Fig. 8 and let  $\chi = S_\nu$ . Suppose  $\mathfrak{s}_1 = \gamma_1 \alpha \eta$  and  $\mathfrak{s}_2 = \gamma_2 \nu \eta \sqrt{m_2 d}$  for some  $\gamma_1, \gamma_2 > 0$  where  $\eta$  is chosen as in Section 2.7.*

*For completeness, let  $m_1, m_2 \geq 512/d$ . Then, the honest prover  $\mathcal{P}$  convinces the honest verifier  $\mathcal{V}$  with probability*

$$\approx 2 \exp\left(\frac{12}{\gamma_1} + \frac{1}{2\gamma_1^2} + \frac{1}{2\gamma_2^2}\right).$$

*For commit-and-prove simulatability, there exists a simulator  $\mathcal{S}$  that, without access to private information  $\mathbf{s}_1, \mathbf{m}$ , outputs a simulation of a commitment  $(\mathbf{t}_A, \mathbf{t}_B)$  along with a non-aborting transcript of the protocol between prover  $\mathcal{P}$  and verifier  $\mathcal{V}$  such that for every algorithm  $\mathcal{A}$  that has advantage  $\varepsilon$  in distinguishing the simulated commitment and transcript from the actual commitment and transcript, whenever the prover does not abort, there is an algorithm  $\mathcal{A}'$  with the same running time that has advantage  $\varepsilon/2 - 2^{-100}$  in distinguishing  $\text{Extended-MLWE}_{n+\ell+\lambda+1, m_2-n-\ell-\lambda-1, \chi, \mathcal{C}, \mathfrak{s}_2}$ .*

*For soundness, denote  $\text{Adv}_{\text{MSIS}}(t)$  to be the maximal probability of a probabilistic algorithm which solves  $\text{MSIS}_{n, m_1+m_2, B}$  and runs in expected time <sup>14</sup> at most  $t$  where  $B := 8\eta\sqrt{\mathfrak{s}_1^2 2m_1 d + \mathfrak{s}_2^2 2m_2 d}$ . Then, there is an extractor  $\mathcal{E}$  with the following properties. When given rewindable black-box access to a probabilistic prover  $\mathcal{P}^*$ , which convinces  $\mathcal{V}$  with probability*

$$\varepsilon > q_1^{-\lambda} + q_1^{-d/l} + 2\text{Adv}_{\text{MSIS}}\left(\frac{9T}{\varepsilon - 2/|\mathcal{C}|}\right) + 2/|\mathcal{C}|, \quad (40)$$

$\mathcal{E}$  outputs  $(\bar{\mathbf{s}}_2, \bar{\mathbf{s}}_1, \bar{\mathbf{m}}) \in \mathcal{R}_q^{m_1+m_2+\ell}$  and  $\bar{c} \in \mathcal{R}_q^\times$  such that

- $\begin{bmatrix} \mathbf{t}_A \\ \mathbf{t}_B \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{0} \end{bmatrix} \cdot \bar{\mathbf{s}}_1 + \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{B} \end{bmatrix} \cdot \bar{\mathbf{s}}_2 + \begin{bmatrix} \mathbf{0} \\ \bar{\mathbf{m}} \end{bmatrix}$
- $f_j((\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]}, (\sigma^i(\bar{\mathbf{m}}))_{i \in [k]}) = 0$  for  $j \in [N]$
- each  $F_u((\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]}, (\sigma^i(\bar{\mathbf{m}}))_{i \in [k]}) \in \mathcal{R}_q$ , where  $u \in [M]$ , has constant coefficient equal to zero
- $\|\bar{c}\|_\infty \leq 2\kappa$

<sup>14</sup> If we assume that the given MSIS problem is computationally hard and  $t$  is polynomial in the security parameter, then  $\text{Adv}_{\text{MSIS}}^t$  must be negligible by the Markov's inequality.

$$- \|\bar{c}\bar{\mathbf{s}}_1\| \leq 2\mathfrak{s}_1\sqrt{2m_1d} \text{ and } \|\bar{c}\bar{\mathbf{s}}_2\| \leq 2\mathfrak{s}_2\sqrt{2m_2d}$$

in expected time at most  $6T/(\varepsilon - 2/|\mathcal{C}|)$  where running  $\mathcal{P}^*$  once is assumed to take at most  $T$  time.

*Remark 4.4.* Let us assume we pick parameters for the proof system such that  $q_1^{-\lambda} + q_1^{-d/l} + 2/|\mathcal{C}|$  is negligible and  $\text{MSIS}_{n,m_1+m_2,B}$  is computationally hard. Suppose there is a prover  $\mathcal{P}^*$  which runs in polynomial time  $T$  and convinces the verifier with noticeable probability  $\varepsilon$ . Then, by our assumption we must have that  $\text{Adv}_{\text{MSIS}}\left(\frac{9T}{\varepsilon - 2/|\mathcal{C}|}\right)$  is negligible. Hence,  $\varepsilon$  satisfies the inequality in Theorem 4.3. Therefore, the extractor  $\mathcal{E}$  will extract the commitment opening and messages in expected polynomial time  $6T/(\varepsilon - 2/|\mathcal{C}|)$ .

*Proof (of Theorem 4.3).* Completeness follows directly from the proof of Theorem 4.2 and the discussion in Section 4.3. As for commit-and-prove simulatability, we simulate the commitment and the transcript identically as in the proof of Theorem 4.2 with two additional steps: (i) we simulate the commitment  $\mathbf{t}_g$  to  $\mathbf{g}$  by setting  $\mathbf{t}_g \leftarrow \mathcal{R}_q^\lambda$  to be a uniformly random vector and (ii) we simulate the polynomials  $h_1, \dots, h_\lambda$  by choosing them uniformly at random from  $X := \{x \in \mathcal{R}_q : \tilde{x} = 0\}$ . Note that we perfectly simulate each  $h_i$  since in the real execution, i.e. (37),  $g_i$ 's are also sampled uniformly from  $X$  and  $\sum_{u=1}^M \gamma_{i,u} F_u((\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]}) \in X$ .

*Knowledge Soundness.* For the remainder of the proof, we focus on the knowledge soundness argument. Informally, we will extract so-called trees of transcripts.

**Definition 4.5 (Tree of transcripts [3]).** Let  $\Pi$  be a  $(2\mu + 1)$ -round public coin protocol, where in the  $2i$ -th round, for  $i \in [\mu]$ , a challenge is chosen from a challenge space  $\mathcal{C}_i$ . A  $(k_1, \dots, k_\mu)$ -tree of transcripts is a set of  $K = \prod_{i=1}^\mu k_i$  transcripts arranged in the following tree structure. The nodes (resp. edges) in this tree correspond to the prover's messages (resp. verifier's challenges). Every node at depth  $i$  has precisely  $k_i$  children corresponding to  $k_i$  pairwise distinct challenges. Every transcript corresponds to exactly one path from the root node to a leaf node.

In our case, the protocol in Fig. 8 is public coin and consists of 7 rounds. We define an extractor  $\mathcal{E}$  as follows. It applies the tree extraction algorithm from [3][Lemma 5] to retrieve a  $(1, 1, 3)$ -tree of transcripts. Then, the expected runtime of  $\mathcal{E}$  is at most  $3T$  and the success probability of  $\mathcal{E}$  is at least  $\varepsilon - 2/|\mathcal{C}|$ .

Using an identical reasoning as in the proof of Theorem 4.2,  $\mathcal{E}$  extracts  $(\bar{\mathbf{s}}_2, \bar{\mathbf{s}}_1, \bar{\mathbf{m}}, \bar{\mathbf{g}}) \in \mathcal{R}_q^{m_1+m_2+\ell+\lambda}$  and  $\bar{c} \in \mathcal{R}_q^\times$  such that

$$\begin{aligned} - \begin{bmatrix} \mathbf{t}_A \\ \mathbf{t}_B \\ \mathbf{t}_g \end{bmatrix} &= \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \cdot \bar{\mathbf{s}}_1 + \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{B} \\ \mathbf{B}_g \end{bmatrix} \cdot \bar{\mathbf{s}}_2 + \begin{bmatrix} \mathbf{0} \\ \bar{\mathbf{m}} \\ \bar{\mathbf{g}} \end{bmatrix} \\ - \sum_{j=1}^{N+\lambda} \mu_j f_j((\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]}, (\sigma^i(\bar{\mathbf{m}} \parallel \bar{\mathbf{g}}))_{i \in [k]}) &= 0 \\ - \|\bar{c}\|_\infty &\leq 2\kappa \\ - \|\bar{c}\bar{\mathbf{s}}_1\| &\leq 2\mathfrak{s}_1\sqrt{2m_1d} \text{ and } \|\bar{c}\bar{\mathbf{s}}_2\| \leq 2\mathfrak{s}_2\sqrt{2m_2d} \end{aligned}$$

or a  $\text{MSIS}_{n,m_1+m_2,B}$  solution. Hence, the probability of getting the former is lower-bounded by

$$\varepsilon - 2/|\mathcal{C}| - \text{Adv}_{\text{MSIS}}(3T) \geq \varepsilon/2 - 1/|\mathcal{C}|.$$

Therefore, by running  $\mathcal{E}$   $1/(\varepsilon/2 - 1/|\mathcal{C}|)$  times, we expect to obtain such a tuple  $(\bar{\mathbf{s}}_2, \bar{\mathbf{s}}_1, \bar{\mathbf{m}}, \bar{\mathbf{g}})$ . The total expected running time is at most

$$3T \cdot \frac{1}{\varepsilon/2 - 1/|\mathcal{C}|} = \frac{6T}{\varepsilon - 2/|\mathcal{C}|}.$$

Now, we turn to proving certain properties of the extracted messages  $(\bar{\mathbf{s}}_1, \bar{\mathbf{m}})$ .

**Lemma 4.6.** Let  $(\bar{\mathbf{s}}_2, \bar{\mathbf{s}}_1, \bar{\mathbf{m}})$  be extracted messages and randomness by  $\mathcal{E}$ . Then, for all  $j = 1, 2, \dots, N$ :

$$f_j((\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]}, (\sigma^i(\bar{\mathbf{m}}))_{i \in [k]}) = 0.$$

*Proof.* Suppose that for some  $j^* \in [N]$  we have:

$$f_{j^*}((\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]}, (\sigma^i(\bar{\mathbf{m}}))_{i \in [k]}) \neq 0.$$

Define  $\mathcal{E}'$  to be a copy of the extractor  $\mathcal{E}$ . Let **success** be the event that  $\mathcal{E}'$  extracts a  $(1, 1, 3)$ -tree of transcripts, where the first message from the prover is denote by  $\mathbf{t}'_g$  and challenges in the second and fourth round are  $(\gamma'_{j,u})$  and  $(\mu'_i)$ , and it also manages to extract  $(\mathbf{s}'_2, \mathbf{s}'_1, \mathbf{m}', \mathbf{g}', \bar{c}')$  such that

$$\begin{aligned} - & \begin{bmatrix} \mathbf{t}_A \\ \mathbf{t}_B \\ \mathbf{t}'_g \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \cdot \mathbf{s}'_1 + \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{B} \\ \mathbf{B}'_g \end{bmatrix} \cdot \mathbf{s}'_2 + \begin{bmatrix} \mathbf{0} \\ \mathbf{m}' \\ \mathbf{g}' \end{bmatrix} \\ - & \sum_{j=1}^{N+\lambda} \mu'_j f_j((\sigma^i(\mathbf{s}'_1))_{i \in [k]}, (\sigma^i(\mathbf{m}' \parallel \mathbf{g}'))_{i \in [k]}) = 0 \\ - & \|\bar{c}'\|_\infty \leq 2\kappa \\ - & \|\bar{c}' \mathbf{s}'_1\| \leq 2\mathbf{s}_1 \sqrt{2m_1 d} \text{ and } \|\bar{c}' \mathbf{s}'_2\| \leq 2\mathbf{s}_2 \sqrt{2m_2 d} \end{aligned}$$

Then, as argued above

$$\varepsilon - 2/|\mathcal{C}| - \text{Adv}_{\text{MSIS}}(3T) \leq \Pr[\text{success}].$$

Now, we focus on upper-bounding the probability of **success**. First, define  $E$  to be the event that<sup>15</sup>

$$\sum_{i=1}^{N+\lambda} \mu'_i f_i((\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]}, (\sigma^i(\bar{\mathbf{m}} \parallel \hat{\mathbf{g}}))_{i \in [k]}) \neq 0$$

where  $\hat{\mathbf{g}} := \mathbf{t}'_g - \mathbf{B}'_g \bar{\mathbf{s}}_2$ . Then, we have  $\Pr[\neg E] \leq q_1^{-d/l}$  since all functions  $f_j$  and variables  $\bar{\mathbf{s}}_1, \bar{\mathbf{m}}$  and  $\hat{\mathbf{g}}$  are independent of the challenges  $(\mu'_j)$ <sup>16</sup> and also

$$f_{j^*}((\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]}, (\sigma^i(\bar{\mathbf{m}} \parallel \hat{\mathbf{g}}))_{i \in [k]}) \neq 0.$$

Next, denote **consistent** to be the event that

$$(\mathbf{s}'_2, \mathbf{s}'_1, \mathbf{m}', \mathbf{g}') = (\bar{\mathbf{s}}_2, \bar{\mathbf{s}}_1, \bar{\mathbf{m}}, \hat{\mathbf{g}}).$$

We concentrate on the term  $\Pr[\text{success} \wedge \neg \text{consistent}]$ . Note that if  $\mathcal{E}'$  succeeds and  $\neg \text{consistent}$  holds, then we have found two different ABDLOP openings to  $(\mathbf{t}_A, \mathbf{t}_B, \mathbf{t}'_g)$ , and thus a  $\text{MSIS}_{n, m_1+m_2, B}$  solution in expected time at most

$$6T/(\varepsilon - 2/|\mathcal{C}|) + 3T \leq 9T/(\varepsilon - 2/|\mathcal{C}|).$$

Therefore

$$\Pr[\text{success} \wedge \neg \text{consistent}] \leq \text{Adv}_{\text{MSIS}}\left(\frac{9T}{\varepsilon - 2/|\mathcal{C}|}\right).$$

All in all, we get

$$\begin{aligned} \Pr[\text{success}] & \leq \Pr[\text{success} \wedge \neg \text{consistent}] + \Pr[\text{success} \wedge \text{consistent}] \\ & \leq \Pr[\text{success} \wedge \text{consistent}|E] + \Pr[\neg E] + \Pr[\text{success} \wedge \neg \text{consistent}] \\ & \leq \Pr[\text{success} \wedge \text{consistent}|E] + q_1^{-d/l} + \text{Adv}_{\text{MSIS}}\left(\frac{9T}{\varepsilon - 2/|\mathcal{C}|}\right). \end{aligned}$$

Next, we focus on  $\Pr[\text{success} \wedge \text{consistent}|E]$ . Note that if  $\mathcal{E}'$  succeeds then we have

$$\sum_{j=1}^{N+\lambda} \mu'_j f_j((\sigma^i(\mathbf{s}'_1))_{i \in [k]}, (\sigma^i(\mathbf{m}' \parallel \mathbf{g}'))_{i \in [k]}) = 0.$$

<sup>15</sup> Note that here  $f_{N+1}, \dots, f_{N+\lambda}$  depend on  $(\gamma'_{j,u})$ .

<sup>16</sup> Indeed, note that  $\hat{\mathbf{g}}$  is independent of  $(\mu'_j)$  since  $\mathbf{t}'_g$  was sent before receiving  $(\mu'_j)$  and  $\bar{\mathbf{s}}_2$  was defined before even running  $\mathcal{E}'$ .

If additionally we have consistency property, i.e.  $(\mathbf{s}'_2, \mathbf{s}'_1, \mathbf{m}', \mathbf{g}') = (\bar{\mathbf{s}}_2, \bar{\mathbf{s}}_1, \bar{\mathbf{m}}, \hat{\mathbf{g}})$ , then we obtain a contradiction assuming that  $E$  is true. Hence,

$$\Pr[\text{success} \wedge \neg \text{consistent} | E] = 0.$$

Therefore,

$$\varepsilon - 2/|\mathcal{C}| - \text{Adv}_{\text{MSIS}}(3T) \leq \Pr[\text{success}] \leq q_1^{-d/l} + \text{Adv}_{\text{MSIS}}\left(\frac{9T}{\varepsilon - 2/|\mathcal{C}|}\right)$$

which leads to contradiction since by assumption

$$\varepsilon > 2\text{Adv}_{\text{MSIS}}\left(\frac{9T}{\varepsilon - 2/|\mathcal{C}|}\right) + q_1^{-d/l} + 2/|\mathcal{C}|.$$

□

Finally, we need to prove that for every  $u = 1, 2, \dots, M$ , the constant coefficient of  $F_u((\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]}, (\sigma^i(\bar{\mathbf{m}}))_{i \in [k]})$  is equal to zero.

**Lemma 4.7.** *Let  $(\bar{\mathbf{s}}_2, \bar{\mathbf{s}}_1, \bar{\mathbf{m}})$  be extracted messages and randomness by  $\mathcal{E}$ . Then, for every  $u = 1, 2, \dots, M$ , the constant coefficient of*

$$F_u((\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]}, (\sigma^i(\bar{\mathbf{m}}))_{i \in [k]})$$

*is equal to zero.*

*Proof.* Suppose that for some  $u^*$ , the constant coefficient of

$$F_{u^*}((\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]}, (\sigma^i(\bar{\mathbf{m}}))_{i \in [k]})$$

is not equal to zero. Similarly as in the proof of the previous lemma, define  $\mathcal{E}'$  to be a copy of the extractor  $\mathcal{E}$ . Let **success** be the event that  $\mathcal{E}'$  extracts a  $(1, 1, 3)$ -tree of transcripts, where the first message by the prover is denoted by  $\mathbf{t}'_g$  and challenges in the second and fourth round are  $(\gamma'_{j,u})$  and  $(\mu'_i)$ , and it extracts  $(\mathbf{s}'_2, \mathbf{s}'_1, \mathbf{m}', \mathbf{g}', \bar{\mathbf{c}}')$  such that

$$\begin{aligned} & - \begin{bmatrix} \mathbf{t}_A \\ \mathbf{t}_B \\ \mathbf{t}_g \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \cdot \mathbf{s}'_1 + \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{B} \\ \mathbf{B}_g \end{bmatrix} \cdot \mathbf{s}'_2 + \begin{bmatrix} \mathbf{0} \\ \mathbf{m}' \\ \mathbf{g}' \end{bmatrix} \\ & - \sum_{j=1}^{N+\lambda} \mu'_j f_j((\sigma^i(\mathbf{s}'_1))_{i \in [k]}, (\sigma^i(\mathbf{m}' \parallel \mathbf{g}'))_{i \in [k]}) = 0 \\ & - \|\bar{\mathbf{c}}'\|_\infty \leq 2\kappa \\ & - \|\bar{\mathbf{c}}'\mathbf{s}'_1\| \leq 2\mathfrak{s}_1\sqrt{2m_1d} \text{ and } \|\bar{\mathbf{c}}'\mathbf{s}'_2\| \leq 2\mathfrak{s}_2\sqrt{2(m_2)d} \end{aligned}$$

As before, we have

$$\varepsilon - 2/|\mathcal{C}| - \text{Adv}_{\text{MSIS}}(3T) \leq \Pr[\text{success}].$$

Define **consistent** to be the event that

$$(\mathbf{s}'_2, \mathbf{s}'_1, \mathbf{m}', \mathbf{g}') = (\bar{\mathbf{s}}_2, \bar{\mathbf{s}}_1, \bar{\mathbf{m}}, \hat{\mathbf{g}})$$

where  $\hat{\mathbf{g}} := \mathbf{t}'_g - \mathbf{B}_g \bar{\mathbf{s}}_2$ . Arguing similarly as for the previous lemma, we have

$$\Pr[\text{success} \wedge \neg \text{consistent}] \leq \text{Adv}_{\text{MSIS}}\left(\frac{9T}{\varepsilon - 2/|\mathcal{C}|}\right).$$

Next, define  $E_1$  to be the event that one of the polynomials  $h_1^*, \dots, h_\lambda^* \in \mathcal{R}_q$  defined as follows:

$$\begin{bmatrix} h_1^* \\ h_2^* \\ \vdots \\ h_\lambda^* \end{bmatrix} := \begin{bmatrix} \hat{g}_1 \\ \hat{g}_2 \\ \vdots \\ \hat{g}_\lambda \end{bmatrix} + \begin{bmatrix} \gamma_{1,1} & \gamma_{1,2} & \cdots & \gamma_{1,M} \\ \vdots & \vdots & \cdots & \vdots \\ \gamma_{\lambda,1} & \gamma_{\lambda,2} & \cdots & \gamma_{\lambda,M} \end{bmatrix} \begin{bmatrix} F_1((\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]}, (\sigma^i(\bar{\mathbf{m}}))_{i \in [k]}) \\ F_2((\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]}, (\sigma^i(\bar{\mathbf{m}}))_{i \in [k]}) \\ \vdots \\ F_M((\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]}, (\sigma^i(\bar{\mathbf{m}}))_{i \in [k]}) \end{bmatrix}$$



has a non-zero constant coefficient. Then, we have  $\Pr[-E_1] \leq q_1^{-\lambda}$  and also

$$\begin{aligned} \Pr[\text{success}] &\leq \Pr[\text{success} \wedge \text{consistent}|E_1] + \Pr[-E_1] + \Pr[\text{success} \wedge \text{-consistent}] \\ &\leq \Pr[\text{success} \wedge \text{consistent}|E_1] + q_1^{-\lambda} + \text{Adv}_{\text{MSIS}} \left( \frac{9T}{\varepsilon - 2/|\mathcal{C}|} \right). \end{aligned}$$

Next, let  $E_2$  be the event that  $h'_1, \dots, h'_\lambda \in \mathcal{R}_q$  output by  $\mathcal{P}^*$  all have constant coefficient equal to zero (which is one of the verification checks). Clearly, we have

$$\begin{aligned} \Pr[\text{success} \wedge \text{consistent}|E_1] &= \Pr[\text{success} \wedge \text{consistent} \wedge E_2|E_1] \\ &\leq \Pr[\text{success} \wedge \text{consistent}|E_1 \wedge E_2]. \end{aligned}$$

Further, define  $E_3$  to be the event that

$$\sum_{j=1}^{N+\lambda} \mu'_j f_j \left( (\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]}, (\sigma^i(\bar{\mathbf{m}} \parallel \hat{\mathbf{g}}))_{i \in [k]} \right) \neq 0.$$

Suppose that events  $E_1$  and  $E_2$  hold. Then, by construction of the quadratic functions  $f_{N+1}, \dots, f_{N+\lambda}$  in (38), there must be some  $u \in [M]$  such that

$$f_{N+u} \left( (\sigma^i(\bar{\mathbf{s}}_1))_{i \in [k]}, (\sigma^i(\bar{\mathbf{m}} \parallel \hat{\mathbf{g}}))_{i \in [k]} \right) \neq 0.$$

Hence,  $\Pr[-E_3|E_1 \wedge E_2] \leq q_1^{-d/l}$  and thus

$$\begin{aligned} \Pr[\text{success} \wedge \text{consistent}|E_1] &\leq \Pr[\text{success} \wedge \text{consistent}|E_3 \wedge E_1 \wedge E_2] + \Pr[-E_3|E_1 \wedge E_2] \\ &\leq \Pr[\text{success} \wedge \text{consistent}|E_3 \wedge E_1 \wedge E_2] + q_1^{-d/l}. \end{aligned}$$

Finally, we claim that  $\Pr[\text{success} \wedge \text{consistent}|E_3 \wedge E_1 \wedge E_2] = 0$ . Indeed, if  $\mathcal{E}'$  succeeds and we have the consistency property then we know that

$$\begin{cases} \sum_{j=1}^{N+\lambda} \mu'_j f_j \left( (\sigma^i(\mathbf{s}'_1))_{i \in [k]}, (\sigma^i(\mathbf{m}' \parallel \mathbf{g}'))_{i \in [k]} \right) = 0 \\ (\mathbf{s}'_2, \mathbf{s}'_1, \mathbf{m}', \mathbf{g}') = (\bar{\mathbf{s}}_2, \bar{\mathbf{s}}_1, \bar{\mathbf{m}}, \hat{\mathbf{g}}) \end{cases}$$

which contradicts the assumption on  $E_3$ . Hence,

$$\Pr[\text{success}] \leq q_1^{-d/l} + q_1^{-\lambda} + \text{Adv}_{\text{MSIS}} \left( \frac{9T}{\varepsilon - 2/|\mathcal{C}|} \right).$$

But we know that the success probability of  $\Pr[\text{success}]$  is at least  $\varepsilon - 2/|\mathcal{C}| - \text{Adv}_{\text{MSIS}}(3T)$ . Therefore, we conclude that

$$\varepsilon \leq q_1^{-d/l} + q_1^{-\lambda} + \text{Adv}_{\text{MSIS}}(3T) + \text{Adv}_{\text{MSIS}} \left( \frac{9T}{\varepsilon - 2/|\mathcal{C}|} \right) + 2/|\mathcal{C}|$$

which leads to contradiction.  $\square$

Eventually, the soundness statement of Theorem 4.3 holds by combining the two previous lemmas.  $\square$

#### 4.4 Reducing the Number of Garbage Commitments

The approach in Section 4.3 requires us to commit to  $\lambda$  additional polynomials  $g_i$ . Here, we consider a special case when  $\sigma := \sigma_{-1}$ <sup>17</sup> and show how to reduce this number by a factor of two for free. In particular, will use the following property of  $\sigma_{-1}$ .

<sup>17</sup> Thus its degree  $k$  is equal to 2.

**Lemma 4.8.** Define the  $\sigma_{-1}$ -trace map  $\text{Tr} : \mathcal{R}_q \mapsto \mathcal{R}_q$  as

$$\text{Tr}(x) = 2^{-1} (x + \sigma_{-1}(x)).$$

Then for any  $a, b \in \mathcal{R}_q$ , the polynomial  $y = \text{Tr}(a) + X^{d/2}\text{Tr}(b)$  satisfies:

$$y_0 = a_0 \text{ and } y_{d/2} = b_0.$$

*Proof.* We first observe that for any  $c \in \mathcal{R}_q$  such that  $\sigma_{-1}(c) = c$  we have  $c_{d/2} = 0$ . Indeed, if we compare the  $d/2$ -th coefficient of  $c$  and  $\sigma_{-1}(c)$ , we get  $c_{d/2} = -c_{d/2}$  and thus  $c_{d/2} = 0$ .

Let  $a' = \text{Tr}(a)$  and  $b' = \text{Tr}(b)$ . Clearly,  $a', b'$  are stable under the  $\sigma_{-1}$  automorphism and hence we have  $a'_{d/2} = b'_{d/2} = 0$ . Also, by construction  $a'_0 = a_0$  and  $b'_0 = b_0$ . Therefore,  $y_0 = a'_0 - b'_{d/2} = a'_0 = a_0$ . Similarly,  $y_{d/2} = a'_{d/2} + b'_0 = b_0$ .  $\square$

For simplicity, suppose that  $\lambda$  is even. The strategy here is to consider each pair  $(a^{(j)}, b^{(j)})_{j \in [\lambda/2]}$  defined as

$$\begin{aligned} a^{(j)} &:= \sum_{u=1}^M \gamma_{2j-1,u} \tilde{f}_u \left( (\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]} \right) \\ b^{(j)} &:= \sum_{u=1}^M \gamma_{2j,u} \tilde{f}_u \left( (\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m}))_{i \in [k]} \right) \end{aligned}$$

and apply Lemma 4.8 to simultaneously prove that the constant coefficient of both elements in  $\mathcal{R}_q$  is equal to zero. Concretely, we prove that the constant *and* middle coefficient of each

$$\text{Tr}(a^{(j)}) + X^{d/2}\text{Tr}(b^{(j)}) \in \mathcal{R}_q$$

is equal to zero.

Similarly as before, we first generate  $\lambda/2$  random masking polynomials  $\mathbf{g} = (g_1, \dots, g_{\lambda/2}) \leftarrow \{x \in \mathcal{R}_q : x_0 = x_{d/2} = 0\}^{\lambda/2}$ . Then, given a challenge matrix  $\Gamma = (\gamma_{i,j}) \leftarrow \mathbb{Z}_q^{\lambda \times M}$ , we construct  $a^{(j)}$  and  $b^{(j)}$  as above and send  $h_1, \dots, h_{\lambda/2}$  defined as follows:

$$h_j = g_j + \text{Tr}(a^{(j)}) + X^{d/2}\text{Tr}(b^{(j)}) \text{ for } j \in [\lambda/2]. \quad (41)$$

The verifier then checks whether the constant and middle coefficient of each  $h_j$  is equal to zero.

Finally, we need to prove that all  $h_1, \dots, h_{\lambda/2}$  are well-formed. As before, our goal will be to define  $\lambda/2$   $k(m_1 + \ell + \lambda/2)$ -variate quadratic functions  $f_{N+1}, \dots, f_{N+\lambda/2} : \mathcal{R}_q^{k(m_1 + \ell + \lambda/2)} \rightarrow \mathcal{R}_q$  such that (41) holds if and only if

$$f_{N+j} \left( (\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m} \parallel \mathbf{g}))_{i \in [k]} \right) = 0 \text{ for } j \in [\lambda/2].$$

First, we observe that:

$$\sigma(\mathbf{s}) = \sigma \left( \begin{bmatrix} \mathbf{s}_1 \\ \sigma(\mathbf{s}_1) \\ \mathbf{m} \\ \sigma(\mathbf{m}) \end{bmatrix} \right) = \begin{bmatrix} \sigma(\mathbf{s}_1) \\ \mathbf{s}_1 \\ \sigma(\mathbf{m}) \\ \mathbf{m} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{I}_{km_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{I}_{km_1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_{k\ell} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{k\ell} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{s}_1 \\ \sigma(\mathbf{s}_1) \\ \mathbf{m} \\ \sigma(\mathbf{m}) \end{bmatrix} = \mathbf{U}\mathbf{s}$$

where  $\mathbf{U} \in \mathcal{R}_q^{k(m_1 + \ell) \times k(m_1 + \ell)}$  is the matrix defined above. Hence, we have the following lemma.

**Lemma 4.9.** Let  $\mathbf{s}_1 \in \mathcal{R}_q^{m_1}$ ,  $\mathbf{m} \in \mathcal{R}_q^\ell$  and set  $\mathbf{s} := (\mathbf{s}_1, \sigma(\mathbf{s}_1), \mathbf{m}, \sigma(\mathbf{m}))$ . For any  $2(m_1 + \ell)$ -variate quadratic function  $f : \mathcal{R}_q^{2(m_1 + \ell)} \rightarrow \mathcal{R}_q$  of the form  $f(\mathbf{x}) = \mathbf{x}^T \mathbf{R}_2 \mathbf{x} + \mathbf{r}_1^T \mathbf{x} + r_0$ , define  $\text{Tr}(f)$  to be the quadratic function

$$\text{Tr}(f)(\mathbf{x}) := \mathbf{x}^T \left( \frac{\mathbf{R}_2 + \mathbf{U}^T \sigma(\mathbf{R}_2) \mathbf{U}}{2} \right) \mathbf{x} + \left( \frac{\mathbf{r}_1^T + \sigma(\mathbf{r}_1^T) \mathbf{U}}{2} \right) \mathbf{x} + \left( \frac{r_0 + \sigma(r_0)}{2} \right).$$

Then, we have  $\text{Tr}(f)(\mathbf{s}) = \text{Tr}(f(\mathbf{s}))$ .

*Proof.* We compute  $\text{Tr}(f(\mathbf{s}))$  from the definition of trace in Lemma 4.8:

$$\begin{aligned} \text{Tr}(f(\mathbf{s})) &= \frac{f(\mathbf{s}) + \sigma(f(\mathbf{s}))}{2} \\ &= \frac{\mathbf{s}^T \mathbf{R}_2 \mathbf{s} + \mathbf{r}_1^T \mathbf{s} + r_0}{2} + \frac{\sigma(\mathbf{s}^T) \sigma(\mathbf{R}_2) \sigma(\mathbf{s}) + \sigma(\mathbf{r}_1^T) \sigma(\mathbf{s}) + \sigma(r_0)}{2} \\ &= \frac{\mathbf{s}^T \mathbf{R}_2 \mathbf{s} + \mathbf{r}_1^T \mathbf{s} + r_0}{2} + \frac{\mathbf{s}^T \mathbf{U} \sigma(\mathbf{R}_2) \mathbf{U} \mathbf{s} + \sigma(\mathbf{r}_1^T) \mathbf{U} \mathbf{s} + \sigma(r_0)}{2} \\ &= \text{Tr}(f)(\mathbf{s}). \end{aligned}$$

Here, we used the observation that  $\sigma(\mathbf{s}) = \mathbf{U} \mathbf{s}$ . □

Let  $\mathbf{x}_1 \in \mathcal{R}_q^{km_1}$ ,  $\mathbf{x}_2 = (\mathbf{x}_{2,1}, \mathbf{x}_{2,2}) \in \mathcal{R}_q^{2(\ell+\lambda)}$ . Denote

$$\mathbf{x}_{2,1} = \left( \mathbf{x}_{2,1}^{(m)}, x_{1,1}^{(g)}, \dots, x_{1,\lambda/2}^{(g)} \right) \text{ and } \mathbf{x}_{2,2} = \left( \mathbf{x}_{2,2}^{(m)}, x_{2,1}^{(g)}, \dots, x_{2,\lambda/2}^{(g)} \right)$$

and set  $\mathbf{x}_2^{(m)} := (\mathbf{x}_{2,1}^{(m)}, \mathbf{x}_{2,2}^{(m)})$ . Then, define

$$\begin{aligned} f_{N+j}(\mathbf{x}_1, \mathbf{x}_2) &:= x_{1,j}^{(g)} + \text{Tr} \left( \sum_{u=1}^M \gamma_{2j-1,u} \tilde{f}_u \right) \left( \mathbf{x}_1, \mathbf{x}_2^{(m)} \right) \\ &\quad + X^{d/2} \text{Tr} \left( \sum_{u=1}^M \gamma_{2j} \tilde{f}_u \right) \left( \mathbf{x}_1, \mathbf{x}_2^{(m)} \right) - h_j. \end{aligned} \tag{42}$$

Then, by Lemma 4.9 we have

$$f_{N+j} \left( (\sigma^i(\mathbf{s}_1))_{i \in [k]}, (\sigma^i(\mathbf{m} \parallel \mathbf{g}))_{i \in [k]} \right) = 0 \text{ for } j \in [\lambda/2]$$

if and only if Equation 41 holds.

As before, in order to prove (34), we define quadratic functions  $f_1, \dots, f_N : \mathcal{R}_q^{2(m_1+\ell+\lambda/2)} \rightarrow \mathcal{R}_q$  as:

$$f_j(\mathbf{x}_1, \mathbf{x}_2) := f_j \left( \mathbf{x}_1, \mathbf{x}_2^{(m)} \right).$$

Finally, we run  $\Pi_{\text{many}}^{(2)} \left( (\mathbf{s}_2, \mathbf{s}_1, \mathbf{m}, \mathbf{g}), \sigma, (f_j)_{j \in [N+\lambda/2]} \right)$  from Fig. 7.

## 5 Applications to Proving Norm Bounds

In this section we provide examples of compound zero-knowledge proofs for various statements based on the protocol in Figure 8. This protocol defined in the previous section proves simultaneously quadratic relations and that the constant coefficient of evaluations of some quadratic functions are 0. We only commit (via ABDLOP) to the messages, but notice that the proven relations may also take as input some automorphisms of these messages. We focus on one specific automorphism to instantiate the general framework of Section 4. Using the notation from Lemma 2.4, this automorphism is  $\sigma := \sigma_{-1}$ . With this choice of automorphism, Lemma 2.5 claims that  $\mathbb{T}$  allows us to prove inner products modulo  $q$  via Figure 8.

### 5.1 Approximate Range Proof

We describe a high-level protocol to prove that a vector  $\mathbf{s}$  is such that  $\|\mathbf{s}\|_2 \leq B$  for some bound  $B$ . The bound we can prove with this method is looser than the actual bound on the norm of  $\mathbf{s}$ , but the counterpart is that the proof is fairly cheap. We will use this protocol to show that when  $\mathbf{s}$  satisfies some relation over

$\mathbb{Z}_q$ , if  $\|\mathbf{s}\|_2 \leq B$  and  $B$  is small enough, then this relation holds over  $\mathbb{Z}$ . The technique is inspired by [23], itself reusing a technique from the  $\ell_\infty$  approximate range proof of [34] adapted to the Euclidean norm.

**Description of the strategy.** The foundation for this protocol is Lemma 2.8. In a nutshell, this Lemma says that for some distribution on the matrix  $R$ , the random projection  $R\vec{s}$  of  $\mathbf{s}$  has approximately the same norm as  $\mathbf{s}$ . This way, we have the opportunity to shrink a potentially very long vector  $\mathbf{s}$  to a much shorter one (e.g length 256) with approximately the same norm. This projection is a  $\mathbb{Z}_q$ -linear map with respect to  $\mathbf{s}$ , which the prover can mask (which entails a slack in the bound we can prove with this method), then send and prove well-formedness of the mask to the verifier.

The matrix  $R$  is a challenge sent by the verifier, and the prover shall prove that  $R\vec{s}$  has small norm so the verifier concludes that so does  $\vec{s}$ . A problem for this method is that the prover cannot reveal the full vector  $R\vec{s}$ , which is why the prover commits to a Gaussian mask  $\mathbf{y}$  of standard deviation  $\mathfrak{s}_3$  for the projection before receiving  $R$ . He applies rejection sampling on  $\vec{z} := \vec{y} + R\vec{s}$ , and computes a zero-knowledge proof of the well-formedness of  $\vec{z}$ . The statement to be proven is captured by Figure 7, and thanks to the rejection sampling step, the  $\vec{z}$  can be revealed to the verifier without leaking information on  $\vec{s}$ . If the well-formedness proof of  $\vec{z}$  checks and  $\|\vec{z}\|$  is small, then it is a matter of parameters for Lemma 2.8 to convince the verifier that  $\vec{s}$  has small norm.

**Bimodal rejection optimization** This mask  $\vec{z}$  of  $R\vec{s}$  is suited to the use of the bimodal trick to reduce the standard deviation  $\mathfrak{s}_3$  of  $\vec{y}$  (therefore also reduce the standard deviation of  $\vec{z}$ , hence the length of the proof). Explicitly, the prover chooses a random sign  $b \in \{-1, 1\}$ , computes  $\vec{z} := bR\vec{s} + \vec{y}$ , and runs the rejection sampling algorithm  $\text{Rej}_0(\vec{z}, bR\vec{s}, \mathfrak{s}_3)$ . The new distribution of  $\vec{z}$  reaches the same number of repetitions as the usual rejection sampling for a lower standard deviation  $\mathfrak{s}_3$ , which shrinks the bit length of  $\vec{z}$ . The extra cost is 1) a commitment to the polynomial  $b$  and 2) a proof that  $b \in \{-1, 1\}$ . The commitment 1) is added to the BDLOP part, and is fairly cheap since  $b$  is a single polynomial. The zero-knowledge proof that  $b$  is a sign 2) comes almost for free as it is a  $\mathbb{Z}_q$ -linear proof amortized with the well-formedness proof of  $\vec{z}$ .

**Proving that a polynomial is a sign.** To perform the bimodal rejection sampling, we need to give a zero-knowledge proof that  $b \in \{-1, 1\}$ . We do this in two steps :

1. We prove that  $b$  is an integer
2. We prove that  $(b - 1)(b + 1) = 0$ .

As  $\mathbb{Z}_q$  is a field, it follows directly from  $(b - 1)(b + 1) = 0$  that  $b$  indeed is a sign.

We prove that  $b$  is an integer by proving that for each polynomial  $\delta_i := X^i \in \mathcal{R}_q$ , the inner product  $\langle \delta_i, b \rangle = 0$ . This inner product maps  $b$  to its  $i$ -th coefficient, and shall therefore be 0 for all positions  $i$  except for the constant coefficient. Second,  $(b - 1)(b + 1) = 0$  is a quadratic function, which we can prove using  $\Pi_{\text{eval}}^{(2)}$  as well. The instantiation of  $\Pi_{\text{eval}}^{(2)}$  is detailed in the next paragraph.

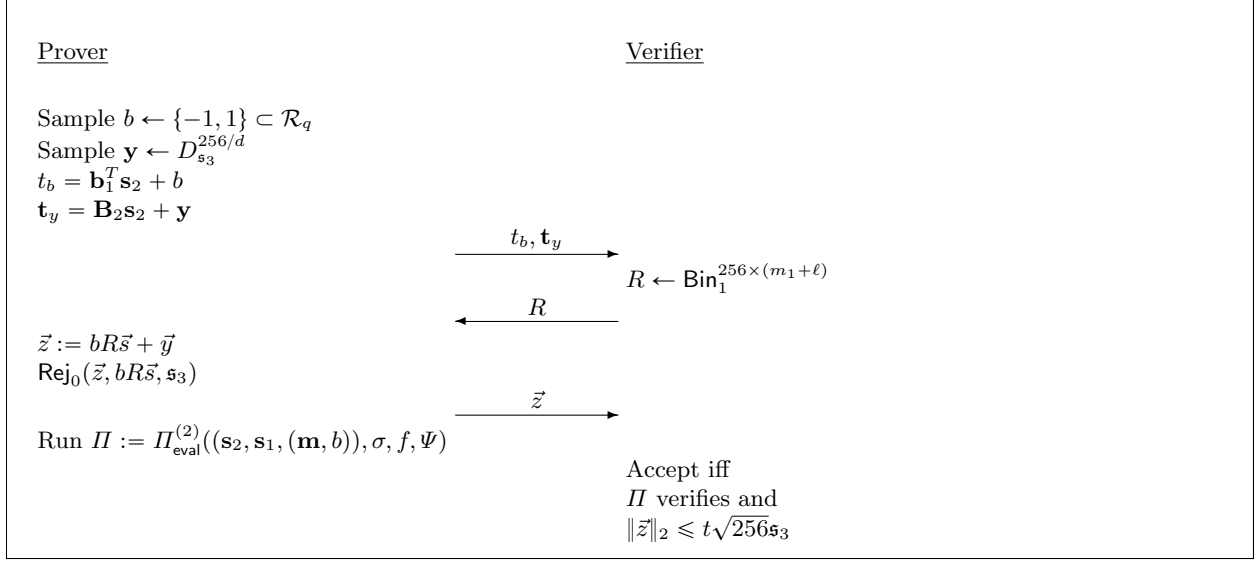
**Instantiation of  $\Pi_{\text{eval}}^{(2)}$**  After 2 rounds, the proof reduces to one amortized zero-knowledge proof for quadratic functions and evaluations. First, the well formedness of the mask  $\mathbf{z}$  of the projection  $R\vec{s}$ , then the proof that  $b$  is a sign. For each of the 256 rows of  $\mathbf{z}$ , we define a function  $F_i$ , and for each of the  $d - 1$  vectors  $\delta_j$ , we define a function  $G_j$ .

$$\begin{aligned} \forall 1 \leq i \leq 256, F_i(\mathbf{s}, \mathbf{y}, b) &= z_i - \mathbb{T}(b\vec{r}_i, \vec{s}) - y_i \\ \forall 1 \leq j \leq d - 1, G_j(b) &= \mathbb{T}(\vec{\delta}_j, \vec{b}), \end{aligned}$$

where  $\vec{r}_i \in \mathbb{Z}_q^{d(m_1 + \ell)}$  is the  $i$ -th row of  $R$ . Finally, to prove that  $b \in \{-1, 1\}$ , we use the functions  $G_j$ 's defined above and the quadratic function  $f(b) = (b - 1)(b + 1)$ . For clarity, we define

$$\Psi = (F_1, \dots, F_{256}, G_1, \dots, G_{d-1}). \quad (43)$$

Private information:  $\mathbf{s} = (\mathbf{s}_1, \mathbf{m}) \in \mathcal{R}_q^{m_1+\ell}$  such that  $\|\mathbf{s}\| \leq \alpha$ , randomness  $\mathbf{s}_2 \in \mathcal{R}_q^{m_2}$ .  
Public information: ABDLOP public parameters and commitment  $\mathbf{t} \in \mathcal{R}_q^{n+\ell}$ ,  
 $\mathbf{b}_1 \in \mathcal{R}_q^{m_2}$ ,  $\mathbf{B}_2 \in \mathcal{R}_q^{(256/d) \times m_2}$ , standard deviation  $\mathfrak{s}_3 := \gamma\sqrt{337}\alpha$ , acceptance coefficient  $t$ .



**Fig. 9:** Commit-and-prove protocol for the messages  $\mathbf{s} = (\mathbf{s}_1, \mathbf{m}) \in \mathcal{R}_q^{m_1+\ell}$ , randomness  $\mathbf{s}_2 \in \mathcal{R}_q^{m_2}$  and  $\bar{\mathbf{c}} \in \bar{\mathcal{C}}$  which satisfy:  $\mathbf{A}_1 \mathbf{s}_1 + \mathbf{A}_2 \mathbf{s}_2 = \mathbf{t}_A$ ,  $\mathbf{B} \mathbf{s}_2 + \mathbf{m} = \mathbf{t}_B$  (ii)  $\|\mathbf{s}_i \bar{\mathbf{c}}\| \leq 2\mathfrak{s}_i \sqrt{2m_i d}$  for  $i = 1, 2$  (where  $\mathfrak{s}_i$  are used in Fig. 6) and (iii)  $\|\mathbf{s}\| \leq 2\sqrt{\frac{256}{26}} t \gamma \sqrt{337} \alpha$ .

**Proposition 5.1.** *The protocol described on Figure 9 is a zero-knowledge proof that the message  $\mathbf{s} = (\mathbf{s}_1, \mathbf{m})$  with  $\|\mathbf{s}\| \leq \alpha$  in the ABDLOP commitment  $\mathbf{t}$  is such that  $\|\mathbf{s}\| \leq 2\sqrt{\frac{256}{26}} t \gamma \sqrt{337} \alpha \leq 189\gamma\alpha$  for  $t = 1.64$ . More precisely, let  $P_{\text{eval}}$  be the success probability of a honest prover in  $\Pi$ ,  $T_{\text{eval}}$  be the run time of the extractor from Theorem 4.3 running on  $\Pi$ . Assume that  $q \geq 41(m_1 + \ell)2\sqrt{\frac{256}{26}} t \gamma \sqrt{337} \alpha$ .*

*For correctness, if the prover and the verifier follow the protocol honestly and  $t = 1.64$ , then the verifier shall accept with probability  $P_{\text{eval}} \cdot (1 - (te^{\frac{1-t^2}{2}})^{256}) \geq P_{\text{eval}} \cdot (1 - 2^{-128})$ .*

*For soundness, let  $\mathcal{P}$  be a probabilistic prover with success probability  $\epsilon$  such that  $\epsilon - 2^{-128}$  verifies Equation (40). There exists an extractor that with rewindable black-box access to  $\mathcal{P}$  recovers a valid opening  $(\bar{\mathbf{s}}_2, \bar{\mathbf{s}}_1, \bar{\mathbf{y}}, \bar{\mathbf{m}}, \bar{b}, \bar{\mathbf{c}})$  to the commitment  $(\mathbf{t}, t_b, \mathbf{t}_y)$  and  $\|(\bar{\mathbf{s}}_1, \bar{\mathbf{m}})\|_2 \leq 2\sqrt{\frac{256}{26}} t \gamma \sqrt{337} \alpha$ , in expected time  $T_{\text{eval}}$ .*

*For commit-and-prove simulatability, there exists a simulator  $\mathcal{S}$  that, without access to private information  $\mathbf{s}_1, \mathbf{m}$ , outputs a simulation of a commitment  $(\mathbf{t}, \mathbf{t}_y, t_b)$  along with a non-aborting transcript of the protocol between prover  $\mathcal{P}$  and verifier  $\mathcal{V}$  such that for every algorithm  $\mathcal{A}$  that has advantage  $\epsilon$  in distinguishing the simulated commitment and transcript from the actual commitment and transcript, whenever the prover does not abort, there is an algorithm  $\mathcal{A}'$  with the same running time that has advantage  $\epsilon/2 - 2^{-100}$  in distinguishing  $\text{Extended-MLWE}_{n+\ell+256/d+2, m_2-n-\ell-256/d, \chi, \mathcal{C}, \mathbf{s}_2}$ .*

*Proof.* We only detail correctness and soundness, commit-and-prove simulatability follows directly from the same property from Figure 8, the rejection sampling and the hiding property of ABDLOP.

**Correctness.** Let  $i \in [256]$ . If the prover and verifier follow the protocol honestly, we have :

$$F_i(\mathbf{s}, \mathbf{y}, b) = z_i - \mathsf{T}(b\vec{r}_i, \vec{s}) - y_i \quad (44)$$

$$\tilde{F}_i(\mathbf{s}, \mathbf{y}, b) = z_i - b\langle \vec{r}_i, \vec{s} \rangle - \vec{y}_i \quad (45)$$

$$= b\langle \vec{r}_i, \vec{s} \rangle - b\langle \vec{r}_i, \vec{s} \rangle = 0. \quad (46)$$

From Equation (44) to Equation (45) comes from Lemma 2.5. Equation (45) to Equation (46) is true because the prover formed  $\vec{z} = R\vec{s} + \vec{y}$  correctly. Obviously, since  $b \in \{-1, 1\}$ ,  $f(b) = 0$ . Again using Lemma 2.5 on the  $G_j$ 's, each functions maps to a non-constant coefficient, which is 0 since in particular  $b \in \mathbb{Z}_q$ . We proved that the inputs of  $\Pi$  are correct, hence with probability  $P_{\text{eval}}$ , the verifier accepts  $\Pi$ . Finally, using the tail bounds from Lemma 2.2 on  $\vec{z}$ , we have that  $\mathbb{P}(\|\mathbf{z}\| \geq t\sqrt{256\mathfrak{s}_3}) \leq (te^{\frac{1-t^2}{2}})^{256}$ , so the verifier also checks  $\|\mathbf{z}\| \leq t\sqrt{256\mathfrak{s}_3}$  with probability at least  $1 - (te^{\frac{1-t^2}{2}})^{256}$ , and the correctness follows.

**Soundness.** The extraction is actually the extraction from the instantiation of  $\Pi_{\text{eval}}^{(2)}$ , and we shall prove that the extracted  $\bar{\mathbf{s}} = (\bar{\mathbf{s}}_1, \bar{\mathbf{m}})$  satisfies  $\|\bar{\mathbf{s}}\| \leq 2\sqrt{\frac{256}{26}}t\gamma\sqrt{337}\alpha$ . We run the extractor from Theorem 4.3. Since we assume that the success probability of the prover verifies Equation (40), the extractor runs in expected time  $T_{\text{eval}}$  and returns  $(\bar{\mathbf{s}}_2, \bar{\mathbf{s}}_1, \bar{\mathbf{y}}, \bar{\mathbf{m}}, \bar{b}) \in \mathcal{R}_q^{m_1+m_2+256/d+\ell+1}$  and  $\bar{c} \in \mathcal{R}_q^\times$  such that  $(\bar{\mathbf{s}}_1, \bar{\mathbf{y}}, \bar{\mathbf{m}}, \bar{b})$  are valid ABDLOP messages for the randomness  $\bar{\mathbf{s}}_2$  and

- $f(\bar{b}) = 0$
- For  $1 \leq i \leq 256$ ,  $\tilde{F}_i(\bar{\mathbf{s}}, \bar{\mathbf{y}}, \bar{b}) = 0$
- For  $1 \leq j \leq d$ ,  $\tilde{G}_j(\bar{b}) = 0$ .

We define  $\bar{\mathbf{s}} = (\bar{\mathbf{s}}_1, \bar{\mathbf{m}})$ . Plugging together the fact that all the  $\tilde{F}_i(\bar{\mathbf{s}}, \bar{\mathbf{y}}, \bar{b})$  are 0 and Equation (45), we have that  $\vec{z}$  is of the correct form, that is  $\vec{z} = R\vec{s} + \vec{y}$ . Under Lemma 2.5,  $\tilde{G}_j(\bar{b}) = 0$  yields that every non-constant coefficient of  $\bar{b}$  is 0, hence  $b \in \mathbb{Z}_q$ . Since  $\mathbb{Z}_q$  is an integral domain we have that  $f(\bar{b}) = (\bar{b} - 1)(\bar{b} + 1) = 0$  ensures that  $\bar{b}$  is a sign.

From the norm verification, we have that

$$\|\mathbf{z}\| \leq t\sqrt{256\mathfrak{s}_3} \quad (47)$$

$$\|R\vec{s} + \vec{y} \pmod{q}\| \leq t\sqrt{256\mathfrak{s}_3} \quad (48)$$

$$\|R\vec{s} + \vec{y} \pmod{q}\| \leq t\sqrt{256}\gamma\sqrt{337}\alpha \quad (49)$$

$$\|R\vec{s} + \vec{y} \pmod{q}\| \leq \frac{1}{2}\sqrt{26} \left( 2\sqrt{\frac{256}{26}}t\gamma\sqrt{337}\alpha \right), \quad (50)$$

where Equation (47) to Equation (48) follows from the proven well-formedness of  $\mathbf{z}$ , Equation (48) to Equation (49) follows from the definition of  $\mathfrak{s}_3$  and Equation (49) to Equation (50) is simply reformulating the upper bound so it fits Lemma 2.8. We now apply Lemma 2.8. Under the condition that  $2\sqrt{\frac{256}{26}}t\gamma\sqrt{337}\alpha \leq \frac{q}{41d(m_1+\ell)}$ , we have that if  $\|\bar{\mathbf{s}}\| \geq 2\sqrt{\frac{256}{26}}t\gamma\sqrt{337}\alpha$ , then the probability over the randomness of the challenge  $R$  that Equation (47) is less than  $2^{-128}$ . By contraposition, with overwhelming probability  $1 - 2^{-128}$ , we have  $\|\bar{\mathbf{s}}\| \leq \sqrt{\frac{256}{26}}t\gamma\sqrt{337}\alpha$ , which completes the soundness proof.

## 5.2 $\ell_2$ -Norm Proof

In this subsection, we describe a general protocol to prove various quadratic relations. We highlight that among the relations this protocol proves is an exact norm proof  $\|\mathbf{s}\| \leq \beta$ , where  $\beta$  is tight. In a nutshell, we prove simultaneously quadratic relations over  $\mathcal{R}_q$ , quadratic relations over  $\mathbb{Z}_q$ , approximate bound on the norm and finally exact bound on the norm. Explicitly, we define public parameters :

- Quadratic functions for  $i \in [\rho]$   $f_i : \mathcal{R}_q^{2(m_1+\ell)} \rightarrow \mathcal{R}_q$
- Evaluation functions for  $i \in [\rho_{\text{eval}}]$   $F_i : \mathcal{R}_q^{2(m_1+\ell)} \rightarrow \mathcal{R}_q$
- For  $i \in [v_d]$ ,  $\mathbf{D}_i \in \mathcal{R}_q^{k_i \times 2(m_1+\ell)}$ ,  $\mathbf{u}_i \in \mathcal{R}_q^{k_i}$
- For  $i \in [v_e]$ ,  $\mathbf{E}_i \in \mathcal{R}_q^{p_i \times 2(m_1+\ell)}$ ,  $\mathbf{v}_i \in \mathcal{R}_q^{p_i}$
- Bounds  $(\beta_i^{(d)})_{i \in [v_d]}, (\beta_i^{(e)})_{i \in [v_e]}$ .

The general statement proven in Figure 10 is the knowledge of a vector  $\mathbf{s} = (\mathbf{s}_1, \sigma(\mathbf{s}), \mathbf{m}, \sigma(\mathbf{m})) \in \mathcal{R}_q^{2m_1} \times \mathcal{R}_q^{2\ell}$  such that

$$\forall 1 \leq i \leq \rho, f_i(\mathbf{s}) = 0 \quad (51)$$

$$\forall 1 \leq i \leq \rho_{\text{eval}}, \tilde{F}_i(\mathbf{s}, \sigma(\mathbf{s})) = 0 \quad (52)$$

$$\forall 1 \leq i \leq v_d, \|\mathbf{D}_i \mathbf{s} - \mathbf{u}_i\|_\infty \leq \beta_i^{(d)}. \quad (53)$$

$$\forall 1 \leq i \leq v_e, \|\mathbf{E}_i \mathbf{s} - \mathbf{v}_i\|_2 \leq \beta_i^{(e)}. \quad (54)$$

The functions  $f_i$  are quadratic relations, and the functions  $F_i$ 's are also quadratic relations but for which we only prove the constant coefficient. The matrices  $\mathbf{D}_i$  and vectors  $\mathbf{u}_i$  are such that  $\mathbf{D}_i \mathbf{s} - \mathbf{u}_i$  is short, and we prove the latter with a looser bound than the actual bound. Finally, the matrices  $\mathbf{E}_i$  and vectors  $\mathbf{v}_i$  are also such that  $\mathbf{E}_i \mathbf{s} - \mathbf{v}_i$  is short, but those we prove in an exact manner.

**General strategy.** Suppose we have an ABDLOP commitment to a vector  $\mathbf{s} = (\mathbf{s}_1, \mathbf{m})$  and we want to prove Equations (51) to (54). To prove the quadratic relations and evaluations Equations (51) and (52), we simply pass on the functions to the input of the instantiation of Figure 8 that we will need later anyway. To prove Equation (53), we use the technique from Figure 9. We now focus on Equation (52). Remind that one can use  $\Pi_{\text{eval}}^{(2)}$  to give a zero-knowledge proof that the inner product of two commitments mod  $q$  is some public constant. Therefore we can prove that  $\langle \mathbf{E}_i \mathbf{s} - \mathbf{v}_i, \mathbf{E}_i \mathbf{s} - \mathbf{v}_i \rangle \pmod q$  is some constant. We use the approximate range proof from Figure 9 to prove that the computation of  $\langle \mathbf{E}_i \mathbf{s} - \mathbf{v}_i, \mathbf{E}_i \mathbf{s} - \mathbf{v}_i \rangle$  does induce a wraparound modulo  $q$ , and therefore also holds over  $\mathbb{Z}$ .

Remember that we do not want to give away the exact norm of  $\mathbf{E}_i \mathbf{s} - \mathbf{v}_i$ , but rather prove that it is lower than some bound. To circumvent this, we prove that the difference between the bound and the norm is a positive integer. Explicitly, we prove that  $(\beta_i^{(e)})^2 - \langle \mathbf{E}_i \mathbf{s} - \mathbf{v}_i, \mathbf{E}_i \mathbf{s} - \mathbf{v}_i \rangle$  can be written with a binary representation  $\vec{x}_i$  of length  $2 \log(\beta_i^{(e)}) \leq d$ <sup>18</sup>. Overall, proving exact norm such as in Equation (54) is the combination of a proof of the relations between  $\mathbf{s}$  and  $(x_i)_{i \in [v_e]}$ , and a proof that each  $x_i$  is binary. Notice that both proofs are over  $\mathbb{Z}$  rather than  $\mathbb{Z}_q$ , so we need a third proof to lift the relations we can only prove directly over  $\mathbb{Z}_q$  to  $\mathbb{Z}$ .

**Proving that a vector is binary.** We detail a strategy to prove that a vector has binary coefficients which improves upon the previous construction from [19]. The strategy relies on the following fact.

**Lemma 5.2.** *Let  $\vec{x} \in \mathbb{Z}_q^{v_e d}$ . If  $\langle \vec{x}, \vec{x} - \vec{1}_{v_e d} \rangle = 0$ , then  $\vec{x} \in \{0, 1\}^{v_e d}$ .*

*Proof.* Every term in the sum is of the form  $a(a-1)$ . One can check that  $a \mapsto a(a-1)$  is a positive function over the integers, therefore, the sum cancels if and only if every term is 0, i.e  $\vec{x}$  is binary.

In other words, it is enough to prove  $\langle \vec{x}, \vec{x} - \vec{1}_{v_e d} \rangle = 0$  to infer that  $\vec{x}$  is a binary vector. We prove the latter in two steps :

1. We prove that  $\langle \vec{x}, \vec{x} - \vec{1}_{v_e d} \rangle = 0 \pmod q$ , which is a direct application of Figure 7.
2. We prove that  $\|\vec{x}\| \leq B$  for some bound  $B$  using Figure 9.

<sup>18</sup> Note that with the parameters we chose,  $(\beta_i^{(e)})^2 - \langle \mathbf{E}_i \mathbf{s} - \mathbf{v}_i, \mathbf{E}_i \mathbf{s} - \mathbf{v}_i \rangle$  is in the range  $[0, \dots, 2^d - 1]$  and therefore  $\vec{x}$  fits in a length  $d$  vector

Provided that  $B$  is such that  $B^2 + \sqrt{v_e d} B < q$  (which is actually very easily met for reasonable parameters),  $\langle \vec{x}, \vec{x} - \vec{1}_{v_e d} \rangle = 0 \pmod q$  holds over the integers, and Lemma 5.2 yields that  $\vec{x}$  is binary.

**Specifications and instantiation.** To begin with, the prover appends a commitment to the binary representation vector  $\mathbf{x} = (x_1 \parallel \dots \parallel x_{v_e})$  in the Ajtai part of the commitment to  $(\mathbf{s}_1, \mathbf{m})$ <sup>19</sup>. This vector is the concatenation of the binary decompositions  $\vec{x}_i$  of  $(\beta_i^{(e)})^2 - \|\mathbf{E}_i \mathbf{s} - \mathbf{v}_i\|_2^2$ . As explained above, this binary decomposition is possible. The verifier samples two approximate range proof challenge matrices  $R^{(d)}, R^{(e)}$ . The first one  $R^{(d)}$  is used for the approximate norm proofs Equation (53) and the second one is used for the exact norm proofs Equation (54). He sends both matrices to the prover. Finally, the prover computes a zero-knowledge proof for the following statements :

$$\forall 1 \leq i \leq \rho, f_i(\mathbf{s}) = 0 \quad (55)$$

$$\forall 1 \leq i \leq \rho_{\text{eval}}, \tilde{F}_i(\mathbf{s}) = 0 \quad (56)$$

$$\forall 1 \leq i \leq v_d, \|\mathbf{D}_i \mathbf{s} - \mathbf{v}_i\|_\infty \leq \beta_i^{(d)} \quad (57)$$

$$\langle \mathbf{x}, \mathbf{x} - \mathbf{1}_{v_e d} \rangle = 0 \pmod q \quad (58)$$

$$\langle \mathbf{E}_i \mathbf{s} - \mathbf{v}_i, \mathbf{E}_i \mathbf{s} - \mathbf{v}_i \rangle + \left( 1 \ 2 \ \dots \ 2^{2 \log(\beta_i^{(e)})} \ 0 \ \dots \ 0 \right) \mathbf{x} = (\beta_i^{(e)})^2 \pmod q \quad (59)$$

$$(\|\mathbf{E}_i \mathbf{s} - \mathbf{v}_i\|_\infty)_{i \in [v_e]}, \|\mathbf{x}\| \text{ are small enough so Equations (58) and (59) hold over } \mathbb{Z}. \quad (60)$$

We proceed to describe the functions in the input of  $\Pi_{\text{eval}}^{(2)}$ . Let us first introduce some notations to make the exposition more compact : we write  $\vec{p}_i = (1 \ 2 \ \dots \ 2^{2 \log(\beta_i^{(e)})} \ 0 \ \dots \ 0)$ ,  $\mathbf{r}_i^j$  the  $j$ -th row of  $R_i$ ,  $y_i^j$  (respectively  $z_i^j$ ) the  $j$ -th coordinate of  $\vec{y}_i$  (respectively  $\vec{z}_i$ ). Remember that  $\forall j \in [d]$ ,  $\delta_j = X^j$  is the unitary monomial of degree  $j$  in  $\mathcal{R}_q$ . We write  $x_i$  the  $i$ -th polynomial of  $\mathbf{x}$ . Finally, we define

$$\mathbf{e}^{(d)} = \begin{bmatrix} \mathbf{D}_1 \mathbf{s} - \mathbf{u}_1 \\ \vdots \\ \mathbf{D}_{v_d} \mathbf{s} - \mathbf{u}_{v_d} \end{bmatrix}, \quad \mathbf{e}^{(e)} = \begin{bmatrix} \mathbf{E}_1 \mathbf{s} - \mathbf{v}_1 \\ \vdots \\ \mathbf{E}_{v_e} \mathbf{s} - \mathbf{v}_{v_e} \\ \mathbf{x} \end{bmatrix}.$$

We define the following functions to instantiate  $\Pi_{\text{eval}}^{(2)}$  :

$$\forall i \in \{d, e\}, g^{(i)}(b^{(i)}) = (b^{(i)} - 1)(b^{(i)} + 1) \quad (61)$$

$$\forall i \in [v_e], G_i(\vec{x}_i) = \mathsf{T}(\vec{x}_i, \vec{x}_i - 1) \quad (62)$$

$$\forall j \in [256], H_j^{(d)}(\mathbf{x}, \mathbf{s}, \mathbf{y}^{(d)}, b^{(d)}) = z_j^{(d)} - \mathsf{T}(b^{(d)} \mathbf{r}_j^{(d)}, \mathbf{e}^{(d)}) - y_j^{(d)} \quad (63)$$

$$\forall j \in [256], H_j^{(e)}(\mathbf{x}, \mathbf{s}, \mathbf{y}^{(e)}, b^{(e)}) = z_j^{(e)} - \mathsf{T}(b^{(e)} \mathbf{r}_j^{(e)}, \mathbf{e}^{(e)}) - y_j^{(e)} \quad (64)$$

$$\forall i \in [v_e], I_i(\mathbf{s}, \mathbf{x}) = \mathsf{T}(\mathbf{E}_i \mathbf{s} - \mathbf{v}_i, \mathbf{E}_i \mathbf{s} - \mathbf{v}_i) + \mathsf{T}(\vec{p}_i, \vec{x}_i) - (\beta_i^{(e)})^2 \quad (65)$$

$$\forall i \in \{d, e\}, 1 \leq j \leq d - 1, J_j^{(i)}(b^{(i)}) = \mathsf{T}(\delta_j, b^{(i)}) \quad (66)$$

We now pack the functions that are the input of  $\Pi_{\text{eval}}^{(2)}$  for more clarity. We let

$$\phi = (f_1, \dots, f_\rho, g^{(d)}, g^{(e)}) \quad (67)$$

$$\Psi = \left( (F_i)_{i \in \rho_{\text{eval}}}, (G_i)_{i \in v_e}, (H_j^{(d)})_{j \in [256]}, (H_j^{(e)})_{j \in [256]}, (I_i)_{i \in v_e}, (J_j^{(i)})_{i \in \{d, e\}, j \in [d]} \right). \quad (68)$$

<sup>19</sup> Note that appending a commitment in the Ajtai part can only be done at the same time as the commitment to  $\mathbf{s}_1$ . If for some reason it is not possible to commit ahead of time to  $\vec{x}$ , one has to commit to  $\vec{x}$  in the BDLOP part instead.



Public information:

Commitment  $\mathbf{t} \in \mathcal{R}_q^{n+\ell}$ ,  $\mathbf{A}_1 \in \mathcal{R}_q^{n \times (m_1+v_e)}$ ,  $\mathbf{A}_2 \in \mathcal{R}_q^{n \times m_2}$ ,  $\mathbf{B} \in \mathcal{R}_q^{\ell \times m_2}$  such that

$$\mathbf{t} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{x} \end{bmatrix} + \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{B} \end{bmatrix} \mathbf{s}_2 + \begin{bmatrix} \mathbf{0} \\ \mathbf{m} \end{bmatrix}. \quad \mathbf{B}^{(d)} \in \mathcal{R}_q^{(256/d) \times m_2}, \mathbf{B}^{(e)} \in \mathcal{R}_q^{(256/d) \times m_2}, \mathbf{b}^{(d)} \in \mathcal{R}_q^{m_2}, \mathbf{b}^{(e)} \in \mathcal{R}_q^{m_2}.$$

For  $i \in [\rho]$ , quadratic functions  $f_i : \mathcal{R}_q^{2(m_1+\ell)} \rightarrow \mathcal{R}_q$

For  $i \in [\rho_{\text{eval}}]$ , quadratic functions  $F_i : \mathcal{R}_q^{2(m_1+\ell)} \rightarrow \mathcal{R}_q$

For  $i \in [v_e]$ , matrix  $(\mathbf{D}_i) \in \mathcal{R}_q^{k_i \times 2(m_1+\ell)}$ , vector  $\mathbf{u}_i \in \mathcal{R}_q^{k_i}$ , bound  $\beta_i^{(d)}$

For  $i \in [v_e]$ , matrix  $(\mathbf{E}_i) \in \mathcal{R}_q^{p_i \times 2(m_1+\ell)}$ , vector  $\mathbf{v}_i \in \mathcal{R}_q^{p_i}$ , bound  $\beta_i^{(e)}$

Vectors  $\mathbf{s}^{(d)} = (\mathbf{D}_1 \mathbf{s} - \mathbf{u}_1 \| \dots \| \mathbf{D}_{v_d} \mathbf{s} - \mathbf{u}_{v_d})$ ,  $\mathbf{s}^{(e)} = (\mathbf{E}_1 \mathbf{s} - \mathbf{v}_1 \| \dots \| \mathbf{E}_{v_e} \mathbf{s} - \mathbf{v}_{v_e} \| \mathbf{x})$ .

Bounds  $\alpha^{(d)} = \|\mathbf{s}^{(d)}\|$ ,  $\alpha^{(e)} = \|\mathbf{s}^{(e)}\|$

Standard deviations  $\mathbf{s}^{(d)} = \gamma^{(d)} \sqrt{337} \alpha^{(d)}$ ,  $\mathbf{s}^{(e)} = \gamma^{(e)} \sqrt{337} (\alpha^{(e)} + \sqrt{d})$ , acceptance coefficient  $t \in \mathbb{R}$

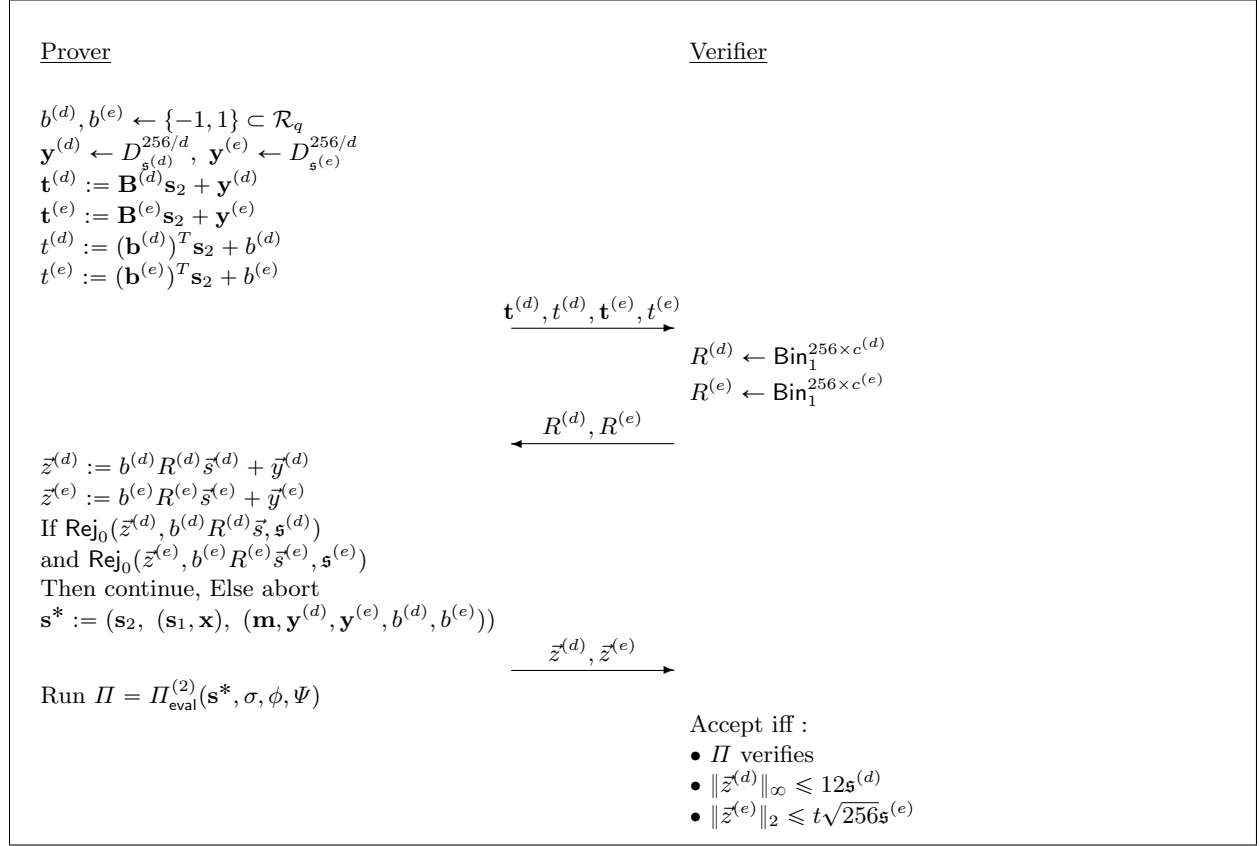
Challenge dimensions  $c^{(d)} = d \sum_{i=1}^{v_e} k_i$ ,  $c^{(e)} = d \sum_{i=1}^{v_e} (p_i + 1)$

Input functions of  $\Pi_{\text{eval}}^{(2)}$   $\phi, \Psi$  defined in Equations (67) and (68).

Private information:

Randomness  $\mathbf{s}_2 \leftarrow \chi^{m_2}$ , message  $\mathbf{s} = (\mathbf{s}_1, \mathbf{m}) \in \mathcal{R}_q^{m_1+\ell}$

such that Equations (51) to (54) hold. Binary decomposition  $x_i \in \mathcal{R}_q$  of  $(\beta_i^{(e)})^2 - \|\mathbf{E}_i \mathbf{s} - \mathbf{v}_i\|^2$ .



**Fig. 10:** Commit-and-prove protocol for messages  $(\mathbf{s}_1, \mathbf{m}) \in \mathcal{R}_q^{m_1+\ell}$ , randomness  $\mathbf{s}_2 \in \mathcal{R}_q^{m_2}$  and  $\bar{c} \in \bar{\mathcal{C}}$  which satisfy:  $\mathbf{A}_1 \mathbf{s}_1 + \mathbf{A}_2 \mathbf{s}_2 = \mathbf{t}_A$ ,  $\mathbf{B} \mathbf{s}_2 + \mathbf{m} = \mathbf{t}_B$  (ii)  $\|\mathbf{s}_i \bar{c}\| \leq 2\mathbf{s}_i \sqrt{2m_i d}$  for  $i = 1, 2$  (where  $\mathbf{s}_i$  are used in Fig. 6) and  $\mathbf{s} := (\mathbf{s}_1, \sigma(\mathbf{s}), \mathbf{m}, \sigma(\mathbf{m}))$  verifies Equations (51) to (54).

**Theorem 5.3.** *The protocol defined on Figure 10 is a commit-and-prove protocol for proving Equations (51) to (54). Concretely, let*

$$B^{(d)} := 24\sqrt{337}\gamma^{(d)}\alpha^{(d)}, \quad B^{(e)} := 2\sqrt{\frac{256}{26}}t\gamma^{(e)}\sqrt{337}(\alpha^{(e)} + \sqrt{v_e d}),$$

$T_{\text{eval}}$  be the run time of the extractor from Theorem 4.3 and  $P_{\text{eval}}$  the probability that a honest prover convinces a honest verifier with the protocol in Figure 8. Assume the following conditions are satisfied :

$$\begin{aligned} B^{(e)} &< \frac{q}{41c^{(e)}} \\ (B^{(e)})^2 + \sqrt{v_e d}B^{(e)} &< q, & 3(\max_{i \in [v_e]} \beta_i^{(e)})^2 + (B^{(e)})^2 - 1 &< q \end{aligned}$$

For correctness, if the prover and the verifier follow the protocol in Figure 10 honestly and  $t = 1.64$ , then the verifier shall accept with overwhelming probability

$$(1 - (te^{\frac{1-t}{2}})^{256})P_{\text{eval}} \geq (1 - 2^{-128})P_{\text{eval}}.$$

For soundness, let  $\mathcal{P}$  be a probabilistic prover with success probability  $\epsilon$  such that  $\epsilon - 2^{128}$  verifies Equation (40). There exists an extractor that with rewindable black-box access to  $\mathcal{P}$ , recovers a valid opening

$$(\bar{\mathbf{s}}_2, (\bar{\mathbf{s}}_1, \bar{\mathbf{x}}), (\bar{\mathbf{m}}, \bar{\mathbf{y}}^{(d)}, \bar{\mathbf{y}}^{(e)}, \bar{b}^{(d)}, \bar{b}^{(e)}), \bar{c}) \in \mathcal{R}_q^{m_1+m_2+\ell+v_d \cdot 256/d+v_e \cdot 256/d+2} \times \mathcal{R}_q^\times$$

for the commitment  $(\mathbf{t}, \mathbf{t}^{(d)}, \mathbf{t}^{(e)}, t^{(d)}, t^{(e)})$  in expected time  $T_{\text{eval}}$ , satisfying Equations (51) to (54).

For commit-and-prove simulatability, there exists a simulator  $\mathcal{S}$  that, without access to private information  $\mathbf{s}_1, \mathbf{m}$ , outputs a simulation of a commitment  $(\mathbf{t}, \mathbf{t}^{(d)}, t^{(d)}, \mathbf{t}^{(e)}, t^{(e)})$  along with a non-aborting transcript of the protocol between prover  $\mathcal{P}$  and verifier  $\mathcal{V}$  such that for every algorithm  $\mathcal{A}$  that has advantage  $\epsilon$  in distinguishing the simulated commitment and transcript from the actual commitment and transcript, whenever the prover does not abort, there is an algorithm  $\mathcal{A}'$  with the same running time that has advantage  $\epsilon/2 - 2^{-100}$  in distinguishing Extended-MLWE $_{n+\ell+2(256/d+1), m_2-n-\ell-2(256/d+1), \chi, \mathcal{C}, \mathbf{s}_2}$ .

*Proof.* We focus on soundness. Correctness follows from the equations detailed in the soundness proof, and commit-and-prove simulatability follows from the commit-and-prove simulatability of Figure 8, the rejection sampling and the hiding property of ABDLOP.

We run the extractor from Theorem 4.3 on  $\Pi$ . Since  $\epsilon$  verifies Equation (40), the extractor of Theorem 4.3 runs in expected time  $T_{\text{eval}}$  and finds

$$(\bar{\mathbf{s}}_2, (\bar{\mathbf{s}}_1, \bar{\mathbf{x}}), (\bar{\mathbf{m}}, \bar{\mathbf{y}}^{(d)}, \bar{\mathbf{y}}^{(e)}, \bar{b}^{(d)}, \bar{b}^{(e)}), \bar{c}) \in \mathcal{R}_q^{m_1+m_2+\ell+v_e \cdot 256/d+v_d \cdot 256/d+2} \times \mathcal{R}_q^\times$$

such that the following holds

- $\forall i \in [\rho], f(\bar{b}) = 0$
- $\forall i \in \{d, e\}, g^{(i)}(\bar{b}^{(i)}) = 0$
- For  $i \in [\rho_{\text{eval}}], \tilde{F}_i(\bar{\mathbf{s}}) = 0$
- For  $i \in [v_e], G_i(\bar{\mathbf{x}}) = 0$
- For  $j \in [256], \tilde{H}_j^{(d)}(\bar{\mathbf{x}}, \bar{\mathbf{s}}, \bar{\mathbf{y}}_d, b_d) = 0$
- For  $j \in [256], \tilde{H}_j^{(e)}(\bar{\mathbf{x}}, \bar{\mathbf{s}}, \bar{\mathbf{y}}_e, b_e) = 0$
- For  $i \in [v_e], \tilde{I}_i(\bar{\mathbf{s}}, \bar{\mathbf{x}}) = 0$
- For  $i \in \{d, e\}, j \in \{1, \dots, d-1\}, \tilde{J}_i(b_i) = 0$ .

We write  $\bar{\mathbf{e}}^{(d)}, \bar{\mathbf{e}}^{(e)}$  the variables  $\mathbf{e}^{(d)}, \mathbf{e}^{(e)}$  where  $\mathbf{s}, \mathbf{x}$  are replaced with their extracted counterparts  $(\bar{\mathbf{s}}_1, \bar{m}), \bar{\mathbf{x}}$ . First,  $\tilde{J}_i(\bar{b}_i) = 0$  yields that all non-constant coefficients of  $\bar{b}^{(d)}$  and  $\bar{b}^{(e)}$  are 0, i.e both  $\bar{b}^{(d)}, \bar{b}^{(e)}$  are constants. Both constants verify  $g^{(i)}(\bar{b}^{(i)}) = 0$ , which under Lemma 2.5 yield that  $\langle \bar{b}^{(d)}, \bar{b}^{(d)} - 1 \rangle \pmod{q} = 0$ , and the same for  $\bar{b}^{(e)}$ . We use Lemma 5.2 to conclude that  $\bar{b}^{(d)}$  and  $\bar{b}^{(e)}$  are signs.

By following the steps from the proof of Proposition 5.1, we jump to the conclusion that, under the condition

$$2\sqrt{\frac{256}{26}}t\gamma\sqrt{337}\alpha^{(e)} \leq \frac{q}{41c^{(e)}},$$

we have with probability  $(1 - 2^{-128})$  that

$$\|\mathbf{e}^{(e)}\| \leq B^{(e)} := 2\sqrt{\frac{256}{26}}t\gamma^{(d)}\sqrt{337}(\alpha^{(e)} + \sqrt{v_e d}). \quad (69)$$

Moreover, from the norm verification we have that  $\|\mathbf{z}^{(d)}\| = \|\mathbf{y}^{(d)} + \bar{b}^{(d)}R^{(d)}\bar{\mathbf{s}}^{(d)}\| \leq 12\mathfrak{s}^{(d)}$ . Notice that  $\bar{b}^{(d)}R^{(d)}$  is distributed according  $\text{Bin}_1^{256}$ , therefore under Lemma 2.6, we conclude that with probability  $1 - 2^{-256}$  we have

$$\|\bar{\mathbf{e}}^{(d)}\| \leq B^{(d)} := 24\mathfrak{s}^{(d)}. \quad (70)$$

We again use Lemma 2.5 to conclude that  $\tilde{G}_i(\bar{\mathbf{s}}, \bar{\mathbf{x}}) = 0$  yields that  $\langle \bar{\mathbf{x}}, \bar{\mathbf{x}} - \mathbf{1} \rangle \pmod{q} = 0$ . Furthermore, we have  $|\langle \bar{\mathbf{x}}, \bar{\mathbf{x}} - \mathbf{1} \rangle| \leq \|\bar{\mathbf{x}}\|_2^2 + \|\bar{\mathbf{x}}\|_1 \leq (B^{(e)})^2 + \sqrt{v_e d}B^{(e)}$ . Since we have  $\|\bar{\mathbf{x}}\|_2 \leq B^{(e)}$  and we assume that  $(B^{(e)})^2 + \sqrt{v_e d}B^{(e)} < q$ , then we obtain  $|\langle \bar{\mathbf{x}}, \bar{\mathbf{x}} - \mathbf{1} \rangle| < q$  over the integers, hence  $\langle \bar{\mathbf{x}}, \bar{\mathbf{x}} - \mathbf{1} \rangle = 0$  over the integers. Lemma 5.2 allows us to infer that  $\bar{\mathbf{x}}$  is indeed binary.

We now proceed to prove the main statements Equations (51) to (54). Since the functions  $(f_i)$  and  $(F_i)$  are passed on directly to  $\Pi$  in Figure 10, Equations (51) and (52) follow directly from the soundness of Figure 8. For Equation (53), we just proved that

$$\|\mathbf{e}^{(d)}\| \leq 24\mathfrak{s}^{(d)},$$

hence for each  $i \in [v_d]$ ,  $\|\mathbf{D}_i\mathbf{s} - \mathbf{u}_i\| \leq 24\sqrt{337}\gamma^{(d)}\alpha^{(d)}$ . Finally, for  $i \in [v_e]$ , we have :

$$\begin{aligned} I_i(\bar{\mathbf{s}}, \mathbf{x}) &= \mathsf{T}(\mathbf{E}_i\bar{\mathbf{s}} - \mathbf{v}_i, \mathbf{E}_i\bar{\mathbf{s}} - \mathbf{v}_i) + \mathsf{T}(\bar{p}_i, x_i) - (\beta_i^{(e)})^2 \\ \widetilde{I}_i(\bar{\mathbf{s}}, \bar{\mathbf{x}}) &= \langle \mathbf{E}_i\bar{\mathbf{s}} - \mathbf{v}_i^T, \mathbf{E}_i\bar{\mathbf{s}} - \mathbf{v}_i^T \rangle + \langle \bar{p}_i, \bar{x}_i \rangle - (\beta_i^{(e)})^2 \pmod{q} \\ &= \|\mathbf{E}_i\bar{\mathbf{s}} - \mathbf{v}_i\|_2^2 + \langle (1 \ 2 \ \dots \ 2^{d-1}), \bar{x}_i \rangle - (\beta_i^{(e)})^2 \\ &= 0, \end{aligned}$$

where we first use Lemma 2.5, then we use the facts that 1) we shown  $\forall i \in [v_e]$ ,  $\|\mathbf{E}_i\bar{\mathbf{s}} - \mathbf{v}_i\| \leq B^{(e)}$  and 2) we assume  $3(\beta_i^{(e)})^2 + (B^{(e)})^2 - 1 < q$ , therefore the whole equation holds over the integers, and finally we use the extracted equation  $\tilde{I}_i(\bar{\mathbf{s}}, \bar{\mathbf{x}}) = 0$ . Rearranging the terms and using the fact that  $\langle \bar{p}_i, \bar{x}_i \rangle \geq 0$ , we get  $\|\mathbf{E}_i\bar{\mathbf{s}} - \mathbf{v}_i\|_2^2 \leq (\beta_i^{(e)})^2$ , which completes the proof.

## 6 Concrete Instantiations

In this section we show how to make use of our techniques for proving norms in the real-world applications, such as proving knowledge of a Module-LWE secret, verifiable encryption and group signatures. In order to show significance of our results, we compare our efficiency with relevant prior work.

## 6.1 General Strategy

We first provide a general strategy on instantiating the protocol in Fig. 8. Firstly, we pick the challenge space  $\mathcal{C}$  as described in Section 2.7. Further, we choose  $\lambda$  and  $l$  such that terms  $q_1^{-\lambda}$  and  $q_1^{-d/l}$  are negligible.

There are two rejection sampling algorithms: one to mask  $\mathbf{cs}_1$  and another one to mask  $\mathbf{cs}_2$ . Denote  $\mathfrak{s}_i = \gamma_i T_i$  where  $T_1, T_2$  are the upper-bounds on  $\|\mathbf{cs}_1\|, \|\mathbf{cs}_2\|$  respectively. The non-aborting probability of the prover is

$$\approx 2 \exp\left(\frac{12}{\gamma_1} + \frac{1}{2\gamma_1^2} + \frac{1}{2\gamma_2^2}\right).$$

Then, as in Section 2.7, we define  $T_1 = \alpha\eta$  and  $T_2 = \nu\eta\sqrt{m_2 d}$ .

Now we set  $n$  and  $m_2$  such that Extended-MLWE and MSIS from Theorem 4.3 are hard against known attacks. We measure the hardness with the root Hermite factor  $\delta$  and aim for  $\delta \approx 1.0044$  similarly as in [12, 4, 19, 34]. We assume that Extended-MLWE is almost as hard as plain MLWE (see [34] for more discussion).

Further, we look at the size of the non-interactive proof outputs via the Fiat-Shamir transform of the protocol in Fig. 8.. First, note that for the non-interactive proof the messages  $\mathbf{w}$  and  $v$  need not be included in the output as they are uniquely determined by the remaining components. Further, the challenges can be generated from a small seed of 256 bits, which itself is generated as the hash of some components. Hence, the contribution of the challenges to the total proof length is very small and thus we neglect it.

As “full-sized” elements of  $\mathcal{R}_q$ , we have  $\mathbf{t}_A, \mathbf{t}_B, \mathbf{t}_g, t$  and  $h_i$ . Therefore, we have in total  $n + \ell + \lambda + 1$  full-sized elements of  $\mathcal{R}_q$ , which altogether costs at most  $(n + \ell + \lambda + 1)d[\log q]$  bits. Now, the only remaining part are the vectors  $\mathbf{z}_1, \mathbf{z}_2$ . Each  $\mathbf{z}_i$  has length  $m_i$  and its coefficients can be bounded by  $12\mathfrak{s}_i$ . Hence, in total, the size of each  $\mathbf{z}_i$  is at most  $m_i d [\log 24\mathfrak{s}_i]$  bits.

In conclusion, the overall commitment and proof length is at most

$$(n + \ell + \lambda + 1)d[\log q] + m_1 d [\log 24\mathfrak{s}_1] + m_2 d [\log 24\mathfrak{s}_2] \text{ bits.} \quad (71)$$

For fair comparison with prior works, we further reduce the commitment and proof size by applying Dilithium [18] compression techniques<sup>20</sup>, as done previously in [34][Appendix B] and [21].

Last but not least, we can reduce the number of garbage terms  $g_j$  from  $\lambda$  to  $\lambda/2$  using the optimisation based on the  $\sigma_{-1}$  automorphism described in Section 4.4.

## 6.2 Proving Knowledge of a Module-LWE Secret

As a primary benchmark for comparison with prior work [19, 34], we prove knowledge of a Module-LWE secret. Namely, we want to prove knowledge of  $(\mathbf{s}, \mathbf{e}) \in \mathcal{R}_q^{M+N}$  such that  $\|(\mathbf{s}, \mathbf{e})\| \leq B$  and

$$\mathbf{A}\mathbf{s} + \mathbf{e} = \mathbf{u} \pmod{q} \quad (72)$$

where  $\mathbf{A} \in \mathcal{R}_q^{N \times M}$  and  $\mathbf{u} \in \mathcal{R}_q^N$  are public.

We propose the following solution using the framework developed in Section 5. Simply, we commit to  $\mathbf{s}_1$  and prove that

$$\left\| \begin{bmatrix} \mathbf{s} \\ \mathbf{A}\mathbf{s} - \mathbf{u} \end{bmatrix} \right\| = \left\| \begin{bmatrix} \mathbf{I}_M \\ \mathbf{A} \end{bmatrix} \mathbf{s} - \begin{bmatrix} \mathbf{0} \\ \mathbf{u} \end{bmatrix} \right\| \leq B.$$

In Fig. 11 we show to properly instantiate the protocol in Fig. 10 to prove knowledge of a Module-LWE secret.

<sup>20</sup> Namely, we do not send the low-order bits of  $\mathbf{t}_A$ . Moreover, if we write  $\mathbf{A}_2 := [\mathbf{A}'_2 \ \mathbf{I}_n]$  and  $\mathbf{s}_2 := \begin{bmatrix} \mathbf{s}_{2,1} \\ \mathbf{s}_{2,2} \end{bmatrix}$  then we do not send the masked opening of  $\mathbf{s}_{2,2}$ . We refer to [34][Appendix B] for more details.

variable	description	instantiation
$\rho$	# of equations to prove	0
$\rho_{\text{eval}}$	# of evaluations with const. coeff. zero	0
$v_e$	# of exact norm proofs	1
$v_d$	# non-exact norm proofs	0
$\mathbf{s}_1$	committed message in the Ajtai part	$\mathbf{s}$
$\mathbf{m}$	committed message in the BDLOP part	$\emptyset$ (no message)
$\mathbf{E}_1$	public matrix for proving $\ \mathbf{E}_1\mathbf{s} - \mathbf{v}_1\  \leq \beta_1$	$\begin{bmatrix} \mathbf{I}_M \\ \mathbf{A} \end{bmatrix}$
$\mathbf{v}_1$	public vector for proving $\ \mathbf{E}_1\mathbf{s} - \mathbf{v}_1\  \leq \beta_1$	$\begin{bmatrix} \mathbf{0} \\ \mathbf{u} \end{bmatrix}$
$\beta_1$	upper-bound on $\ \mathbf{E}_1\mathbf{s} - \mathbf{v}_1\  \leq \beta_1$	$B$

**Fig. 11:** Instantiation of the protocol in Fig. 10 for proving  $\mathbf{A}\mathbf{s} + \mathbf{e} = \mathbf{u} \pmod{q}$  and  $\|(\mathbf{s}, \mathbf{e})\| \leq B$ . The variables in the first two columns refer to the ones defined in Section 5 and the ones in the last column refer to the parameters in this subsection.

*Remark 6.1.* We note that [19, 34] could not avoid committing to  $\mathbf{e}$  without having additional commitments. Indeed, previous work efficiently prove smallness of a vector  $\mathbf{s}$ , e.g.  $\|\mathbf{s}\|_\infty \leq 1$ , by committing to its coefficient vector  $\vec{s}_1$  using NTT slots and then proving that  $\vec{s} \circ (\vec{s} - \vec{1}) \circ (\vec{s} + \vec{1}) = \vec{0}$  [4]. If one were not to commit to  $\mathbf{e}$ , then one would need to prove an equation of the form

$$(A\vec{s} - \vec{u}) \circ (A\vec{s} - \vec{u} - \vec{1}) \circ (A\vec{s} - \vec{u} + \vec{1}) = \vec{0}.$$

However, this relation, which is a mix of linear and product relations, cannot be proven using current methods included in [19, 34] without making intermediate commitments.

**Parameters.** We instantiate our protocol for the case when  $q \approx 2^{32}$  and  $N = M = 1024/d$  similarly as in [12, 19, 34] using the methodology in Section 6.1. We provide a summary of our parameter selection in Table 12.

Let us pick prime  $q := 2^{32} - 2735$  (i.e.  $q = q_1$ ) and set  $d = 128, l = 8$  and  $B = \sqrt{2048}^{21}$ . Then we define the randomness distribution  $\chi$  as a uniform one over  $S_1$ . For the challenge space, we set  $\kappa = 2, \eta = 72$  as in Fig. 3. Also, for  $q \approx 2^{32}$ , we choose  $l = 4$  and  $\lambda = 4$ . Then,  $q^{-d/l} < q^{-\lambda} < 2^{-128}$  and  $\kappa = 2 < 2.82 \approx \frac{1}{2\sqrt{l}}q^{1/l}$ .

There are three rejection sampling algorithms: one to mask  $c\mathbf{s}_1$ , another one to mask  $c\mathbf{s}_2$  and the last one to mask  $\|R\vec{s}_1\|$ . Denote  $\mathfrak{s}_i = \gamma_i T_i$  where  $T_1, T_2, T_3$  are the upper-bounds on  $\|c\mathbf{s}_1\|, \|c\mathbf{s}_2\|$  and  $\|R\vec{s}_1\|$  respectively. The repetition rate in our case is at least

$$2 \exp\left(\frac{12}{\gamma_1} + \frac{1}{2\gamma_1^2} + \frac{1}{2\gamma_2^2} + \frac{1}{2\gamma^{(e)^2}}\right).$$

The rate in [34] is around 7 hence we set  $\gamma_1 = 11, \gamma_2 = 1.85$  and  $\gamma^{(e)} = 5$ .

We define  $T_1 = \alpha\eta$  and  $T_2 = \eta\sqrt{m_2d}$  as explained in Section 6.1 where  $\alpha = \sqrt{B^2 + d}$ . For  $T_3$  we use Lemma 2.7 and set  $T_3 = \sqrt{337}\alpha$ .

The total communication size has been already analysed in Section 6.1 with the only addition being the vector  $\mathbf{z}$  sent in the third round in Fig. 9. Its size is at most  $256 \log(24\mathfrak{s}_3)$  bits which is significantly small compared to the other proof components. Hence, the total commitment and proof size is around 14.4KB.

### 6.3 Verifiable Encryption

For presentation, we will consider a standard Regev public-key encryption scheme [39] but similar analysis can be applied for more complex construction, such as Kyber [13], Saber [15] and NTRU [24] (see [32][Section 4]

<sup>21</sup> It is the case when  $\mathbf{s}_1, \mathbf{e}$  only consist of ternary coefficients as assumed in the prior work.

parameters	description	value
$q$	prime modulus	$2^{32} - 2735$
$d$	ring dimension for of $\mathcal{R}$	128
$l$	# factors $X^d + 1$ splits into mod $q$	4
$N$	height of the $\mathbf{A}$ matrix	8
$M$	width of the $\mathbf{A}$ matrix	8
$\gamma_1$	rejection sampling constant for $cs_1$	11
$\gamma_2$	rejection sampling constant for $cs_2$	1.85
$\gamma^{(e)}$	rejection sampling constant for the ARP	5
$\kappa$	maximum coefficient of a challenge in $\mathcal{C}$	2
$n$	height of matrices $\mathbf{A}_1, \mathbf{A}_2$ in ABDLOP	10
$m_1$	length of the message $\mathbf{s}_1$ in the “Ajtai” part	8
$\ell$	length of the message $\mathbf{m}$ in the “BDLOP” part	0
$\lambda$	number of garbage $g_j \in \mathcal{R}_q$ for boosting soundness	4
$m_2$	length of the randomness $\mathbf{s}_2$ in ABDLOP	26
$\nu$	randomness $\mathbf{s}_2$ is sampled from $S_\nu^{m_2}$	1
	repetition rate	7
	commitment + proof size	14.4KB

**Fig. 12:** Parameter selection for proving  $\mathbf{As} + \mathbf{e} = \mathbf{u} \pmod{q}$  and  $\|(\mathbf{s}, \mathbf{e})\| \leq \sqrt{2048}$  using the protocol in Fig. 10

for more details). Namely, let  $p$  be a prime modulus of the encryption scheme. In order to encrypt a binary message  $m \in \{0, 1\}^d$  with  $\omega$  number of 1s<sup>22</sup>, a user samples a randomness vector  $\mathbf{r} \leftarrow \xi^k$ , where  $\xi$  is a distribution over  $\mathcal{R}$ , and compute

$$\begin{bmatrix} t_0 \\ t_1 \end{bmatrix} := \begin{bmatrix} \mathbf{A} \\ \mathbf{b}^T \end{bmatrix} \mathbf{r} + \begin{bmatrix} \mathbf{0} \\ \lfloor \frac{p}{2} \rfloor m \end{bmatrix} \quad (73)$$

over  $\mathcal{R}_p := \mathbb{Z}_p[X]/(X^d + 1)$  where  $(\mathbf{A}, \mathbf{b}) \in \mathcal{R}_p^{N \times K} \times \mathcal{R}_p^N$  is the public key<sup>23</sup>. Let  $B$  be an upper-bound on  $\mathbf{r}$  such that the probability that  $\|\mathbf{r}\| > B$  for  $\mathbf{r} \leftarrow \xi^K$  is negligible. Then, in the verifiable encryption scenario, we want to prove knowledge of  $\mathbf{r} \in \mathcal{R}^K$  and  $m \in \mathcal{R}$  such that (i) Equation 73 is satisfied over  $\mathcal{R}_p$ , (ii)  $\|\mathbf{r}\| \leq B$  and (iii)  $m \in \{0, 1\}^d$  and  $\|m\|_2 = \sqrt{\omega}$ .

The high-level idea is to commit to  $(\mathbf{r}, m)$  using the ABDLOP commitment modulo  $q$  and prove these three statements. Note that the latter two have already been covered in Section 5. Hence, from now on we focus on proving the first statement.

We first observe that if  $q$  is divisible by  $p$  then (73) can be transformed into a linear equation modulo  $q$  and can be proven as described in Section 4. However, in practical instantiations  $p$  will be significantly small relative to  $q$  (e.g.  $p = 3329$  in Kyber). Consequently, if  $q$  has a small prime divisor  $p$  then by Theorem 4.3, we would need to commit to more garbage polynomials  $g_i$  in order to keep the soundness error negligible. Moreover, for implementation purposes one might want  $p$  to be a prime such that  $X^d + 1$  splits into many factors modulo  $p$  (e.g.  $p = 3329$ ). In this case, if  $p$  divides  $q$ , then the challenge space  $\mathcal{C}$  does not have the invertibility property which is necessary for the soundness proof. In Fig. 14 we propose an example instantiation for the case when  $q$  is divisible by  $p$  (see parameter set II).

Now, suppose that  $p$  is co-prime to  $q$ . Then, (73) is true if and only if there exists a vector  $\mathbf{v} \in \mathcal{R}^{N+1}$  such that

$$\begin{bmatrix} t_0 \\ t_1 \end{bmatrix} := \begin{bmatrix} \mathbf{A} \\ \mathbf{b}^T \end{bmatrix} \mathbf{r} + \begin{bmatrix} \mathbf{0} \\ \lfloor \frac{p}{2} \rfloor m \end{bmatrix} + p\mathbf{v} \quad (74)$$

<sup>22</sup> Hence, the message space is  $\binom{d}{\omega}$ .

<sup>23</sup> Recall that all coefficients of the terms involved in (73) are between  $-p/2$  and  $p/2$ .

over  $\mathcal{R}$ . From a simple calculation,  $\|\mathbf{v}\|_\infty \leq B\sqrt{kd}/2 + 1$ . We can avoid committing to  $\mathbf{v}$ , similarly as in Section 6.5, by proving directly that vector

$$\mathbf{v} := p^{-1} \cdot \left( \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{b}^T & \lfloor \frac{p}{2} \rfloor \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ m \end{bmatrix} - \begin{bmatrix} \mathbf{t}_0 \\ t_1 \end{bmatrix} \right) \in \mathcal{R}_q^n \quad (75)$$

has norm at most  $B_v := (B\sqrt{Kd}/2 + 1)\sqrt{(N+1)d}$ . Since this expression is linear in the committed messages  $\mathbf{r}$  and  $m$ , we can apply the protocol in Fig. 10 to prove its norm. As we will show below, it is enough to prove an approximate bound, i.e.  $\|\mathbf{v}\|_\infty \leq B_v \cdot \psi$ , where  $\psi := 2 \cdot 12 \cdot \gamma^{(d)} \cdot \sqrt{337}$ , as described in Section 5. Indeed, in the soundness argument we would extract a pair  $(\mathbf{r}^*, m^*)$  which satisfies

$$\begin{cases} m^* \in \{0, 1\}^d, \\ \|\mathbf{r}^*\| \leq B, \\ \left\| p^{-1} \cdot \left( \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{b}^T & \lfloor \frac{p}{2} \rfloor \end{bmatrix} \begin{bmatrix} \mathbf{r}^* \\ m^* \end{bmatrix} - \begin{bmatrix} \mathbf{t}_0 \\ t_1 \end{bmatrix} \right) \right\|_\infty \leq B_v \psi. \end{cases}$$

Denote the third expression as  $\mathbf{v}^* \in \mathcal{R}^{N+1}$ . Then, we have

$$\begin{bmatrix} \mathbf{t}_0 \\ t_1 \end{bmatrix} \equiv \begin{bmatrix} \mathbf{A} \\ \mathbf{b}^T \end{bmatrix} \mathbf{r}^* + \begin{bmatrix} \mathbf{0} \\ \lfloor \frac{p}{2} \rfloor m^* \end{bmatrix} + p\mathbf{v}^* \pmod{q}. \quad (76)$$

Thus,

$$\left\| \begin{bmatrix} \mathbf{A} \\ \mathbf{b}^T \end{bmatrix} \mathbf{r}^* + \begin{bmatrix} \mathbf{0} \\ \lfloor \frac{p}{2} \rfloor m^* \end{bmatrix} + p\mathbf{v}^* - \begin{bmatrix} \mathbf{t}_0 \\ t_1 \end{bmatrix} \right\|_\infty \leq p \left( B\sqrt{Kd}/2 + 1 + B_v \psi \right).$$

Hence, if  $q$  is bigger than the right-hand side of this inequality, then we conclude that Equation (76) holds over integers. In particular  $(\mathbf{t}_0, t_1)$  is a valid encryption of  $m$  under randomness  $\mathbf{r}$  over  $\mathcal{R}_p$ .

In Fig. 13 we instantiate the protocol from Fig. 10 for verifiable encryption as described above.

*Remark 6.2.* Note that the current state-of-the-art lattice based verifiable encryption [31], which is used in e.g. [16, 32], only provide *relaxed* verifiable encryption. Namely, the soundness argument only guarantees knowledge of a message and randomness corresponding to the ciphertext  $(\bar{c}\mathbf{t}_0, \bar{c}t_1)$ , where  $\bar{c} \in \mathcal{R}_p$  is called a relaxation factor. More importantly,  $\bar{c}$  is not known to the decryptor and thus it guesses a  $\bar{c}$  and attempts to recover the ciphertext  $(\bar{c}\mathbf{t}_0, \bar{c}t_1)$ . Consequently, the prior works had to equate the decryption time with the adversary's running time. Here, since we commit to  $\mathbf{r}$  and  $m$  using a separate ABDLOP commitment, we circumvent the relaxation factor by proving exact norms on  $\mathbf{r}$  and  $m \in \{0, 1\}^d$ .

**Parameters.** We provide our parameters choices<sup>24</sup> in Fig. 14. For the ciphertext modulus and dimensions, we follow the Kyber instantiation. In particular,  $N = 4, K = 9$  and  $\mathbf{b} = \mathbf{A}^T \mathbf{s} + \mathbf{e}$  where the secret key  $\mathbf{s}$  and error  $\mathbf{e}$  come from  $\text{Bin}_2^{4d}$  and  $\text{Bin}_2^{Kd}$  respectively. For the randomness distribution  $\xi := \text{Bin}_2^d$ . Hence, we can set the upper-bound  $B$  on the norm of  $\mathbf{r} \leftarrow \xi^K$  as  $B = 2\sqrt{Kd}$  and thus  $B_v = (Kd + 1)\sqrt{(N+1)d}$ . We checked that the decryption failure probability for these parameters is at most  $2^{-300}$ .

The rest of the parameters are chosen similarly as in Sections 6.1 and 6.2. Finally, we need to check that

$$q \approx 2^{36} > p \cdot \left( B\sqrt{Kd}/2 + 1 + (B\sqrt{Kd}/2 + 1)\sqrt{(N+1)d}\psi \right).$$

The term on the right-hand side is less than  $2^{36}$  thus the inequality holds.

<sup>24</sup> One can also instantiate the encryption scheme over a larger ring, e.g.  $\mathcal{R}' := \mathbb{Z}[X]/(X^{256} + 1)$ . Then, in order to apply our proof system over a smaller ring  $\mathcal{R}$ , one would first map the equations to work over  $\mathcal{R}$  rather than  $\mathcal{R}'$  as described in [32][Section 2.8].

variable	description	instantiation
$\rho$	# of equations to prove	0
$\rho_{eval}$	# of evaluations with const. coeff. zero	2
$v_e$	# of exact norm proofs	1
$v_d$	# non-exact norm proofs	1
$\mathbf{s}_1$	committed message in the Ajtai part	$(\mathbf{r}, m)$
$\mathbf{m}$	committed message in the BDLOP part	$\emptyset$ (no message)
$F_1$	evaluation to prove const coeff. zero	$\sigma_{-1}(m) \left( m - \sum_{i=0}^{d-1} X^i \right)$
$F_2$	evaluation to prove const coeff. zero	$\sigma_{-1}(m) \cdot m - \omega$
$\mathbf{E}_1$	public matrix for proving $\ \mathbf{E}_1 \mathbf{s} - \mathbf{v}_1\  \leq \beta_1^{(e)}$	$\mathbf{I}_{K+1}$
$\mathbf{v}_1$	public vector for proving $\ \mathbf{E}_1 \mathbf{s} - \mathbf{v}_1\  \leq \beta_1^{(e)}$	$\mathbf{0}$
$\beta_1^{(e)}$	upper-bound on $\ \mathbf{E}_1 \mathbf{s} - \mathbf{v}_1\  \leq \beta_1^{(e)}$	$\sqrt{B^2 + \omega}$
$\mathbf{D}_1$	public matrix for proving $\ \mathbf{D}_1 \mathbf{s} - \mathbf{u}_1\  \leq \beta_1^{(d)}$	$p^{-1} \cdot \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{b}^T & \lfloor \frac{p}{2} \rfloor \end{bmatrix}$
$\mathbf{u}_1$	public vector for proving $\ \mathbf{D}_1 \mathbf{s} - \mathbf{u}_1\  \leq \beta_1^{(d)}$	$p^{-1} \cdot \begin{bmatrix} \mathbf{t}_0 \\ t_1 \end{bmatrix}$
$\beta_1^{(d)}$	upper-bound on $\ \mathbf{D}_1 \mathbf{s} - \mathbf{u}_1\  \leq \beta_1^{(d)}$	$(B\sqrt{Kd}/2 + 1)\sqrt{(N+1)d}$

**Fig. 13:** Instantiation of the protocol in Fig. 10 for verifiable encryption. The variables in the first two columns refer to the ones defined in Section 5 and the ones in the last column refer to the parameters in this subsection. Functions  $F_1, F_2$  are used to prove that  $m$  has binary coefficients and  $\|m\| = \sqrt{\omega}$ . Then, a triple  $(\mathbf{E}_1, \mathbf{v}_1, \beta_1^{(e)})$  corresponds to proving exactly that  $\|(\mathbf{r}, m)\| \leq \sqrt{B^2 + \omega}$ . Since we know  $\|m\| = \sqrt{\omega}$ , this implies that  $\|\mathbf{r}\| \leq B$ . The last triple  $(\mathbf{D}_1, \mathbf{u}_1, \beta_1^{(d)})$  corresponds to proving approximately that  $\|\mathbf{v}\| \leq (B\sqrt{Kd}/2 + 1)\sqrt{(N+1)d}$  where  $\mathbf{v}$  is defined in (75).

## 6.4 Group Signature

We apply our proof system to the recent group signature construction by Lyubashevsky et al. [32]. Our construction inherits a big advantage from [16, 32], namely signature generation and verification time do not depend on the size of the group. We first sketch the scheme and refer to [32] for more details. In this subsection, we work over the larger ring  $\mathcal{R}_{kd} := \mathbb{Z}[X]/(X^{kd} + 1)$  where  $k \geq 1$  is a power-of-two. Then, define  $\mathcal{R}_{kd,p} := \mathcal{R}_{kd}/(p)$  for an integer  $p$ . The benefit of having a larger ring than  $\mathcal{R}$  is small public key size of our group signature. Operations in the construction will be over  $\mathcal{R}_{kd,p}$  where  $p$  is prime.

**Overview.** Let  $G \subseteq \mathcal{R}_{kd,p}$  be the identity space. The group manager first samples  $\mathbf{A} \leftarrow \mathcal{R}_{kd,p}^{N \times (N+M)}$ ,  $\mathbf{B}' \leftarrow \mathcal{R}_{kd,p}^{N \times \tau N}$ , randomness matrix  $\mathbf{R} \leftarrow S_{kd,1}^{(N+M) \times \tau N}$ , where

$$S_{kd,1} := \{x \in \mathcal{R}_{kd} : \|x\|_\infty \leq 1\}$$

and sets  $\mathbf{B} := \mathbf{A}\mathbf{R}$ . Further, it samples  $\mathbf{u} \leftarrow \mathcal{R}_{kd,p}^N$ . Then, the public key is a tuple

$$gpk := (\mathbf{A}, \mathbf{B}, \mathbf{B}', \mathbf{u}).$$

Now, for each user with identity  $i \in G$ , the group manager samples the secret key

$$sk_i := (\mathbf{s}_1^{(i)}, \mathbf{s}_2^{(i)}, \mathbf{s}_3^{(i)}) \leftarrow D_s^{((2\tau+1)N+M)kd}$$

such that

$$[\mathbf{A} | \mathbf{B} + i\mathbf{G} | \mathbf{B}'] \begin{bmatrix} \mathbf{s}_1^{(i)} \\ \mathbf{s}_2^{(i)} \\ \mathbf{s}_3^{(i)} \end{bmatrix} = \mathbf{u}$$



parameter set	description	I	II
$p$	encryption modulus	3329	3373
$N$	height of $\mathbf{A}$	4	4
$K$	width of $\mathbf{A}$	9	9
$\xi$	$\xi^K$ is the randomness distribution of $\mathbf{r}$	$\text{Bin}_2^d$	$\text{Bin}_2^d$
$q$	proof system modulus	$\approx 2^{36}$	$\approx 2^{32}$
$d$	dimension of $\mathcal{R}$	128	128
$l$	# factors $X^d + 1$ splits into mod $q$	8	2
$\gamma_1$	rej. samp. constant for $\mathbf{cs}_1$	11	11
$\gamma_2$	rej. samp. constant for $\mathbf{cs}_2$	1.6	0.9
$\gamma^{(e)}$	rej. samp. constant for exact ARP	12	2.5
$\gamma^{(d)}$	rej. samp. constant for non-exact ARP	1	–
$\kappa$	maximum coefficient of a challenge in $\mathcal{C}$	2	2
$n$	height of $\mathbf{A}_1, \mathbf{A}_2$ in ABDLOP	10	10
$m_1$	length of the “Ajtai” message $\mathbf{s}_1$	10	6
$\ell$	length of the “BDLOP” message $\mathbf{m}$	0	0
$\lambda$	# of garbage $g_j$ for soundness	4	12
$m_2$	length of randomness $\mathbf{s}_2$	31	30
$\nu$	randomness $\mathbf{s}_2$ is sampled from $S_\nu^{m_2}$	1	1
	repetition rate	12	12
	ciphertext size	1KB	1KB
	commitment + proof size	19.5KB	18.3KB

**Fig. 14:** Parameter selection, ciphertext and proof sizes for verifiable encryption. For the second parameter set we choose  $q := 1273301 \cdot 3373$ . Since  $p$  divides  $q$ , we do not need to do an approximate range proof of  $\mathbf{v}$  as for I. Consequently, we can pick smaller modulus  $q$  and apply a similar strategy as in Section 6.2.

using the [38] trapdoor sampling with standard deviation  $\mathfrak{s}$  where  $\mathbf{G} := \mathbf{I}_N \otimes [1 \ g \ \cdots \ g^{\tau-1}]$  is a gadget matrix and  $g := \lfloor p^{1/\tau} \rfloor$ .

The high level idea for signing is for the user with identity  $i \in G$  to prove knowledge of  $i$  and their secret key  $sk_i := (\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3) \in \mathcal{R}_{kd,p}^{(2\tau+1)N+M}$  which satisfy:

$$[\mathbf{A}|\mathbf{B} + i\mathbf{G}|\mathbf{B}'] \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \\ \mathbf{s}_3 \end{bmatrix} = \mathbf{u}, \quad \left\| \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \\ \mathbf{s}_3 \end{bmatrix} \right\| \leq B := \mathfrak{s} \sqrt{2((2\tau+1)N+M)kd}, \quad i \in G. \quad (77)$$

For the bound  $B$  we used Lemma 2.2 for  $t = \sqrt{2}$ .

In order to be able to open the group signature scheme, we will add a verifiable encryption to the signature. Namely, we want the signer to encrypt their identity  $i$ , using a public key associated to a decryption key that the group manager possesses, and prove that this encryption is indeed of their identity. We do this exactly as described in Section 6.3 with a prime  $p_{\text{enc}} := 3329$ . Similarly, all the dimensions and bounds included in that Section will be written with subscript  $\text{enc}$ .

**Efficient Proof of (77).** To begin with, note that relations over  $\mathcal{R}_{kd,p}$  such as the first one in Equation (77) can be written equivalently over our usual subring  $\mathcal{R}_p$ . Indeed, Lyubashevsky et al. showed that there is an efficiently computable ring isomorphism between  $\mathcal{R}_{kd}$  and  $\mathcal{R}^k$ , for an appropriately defined vector multiplication in  $\mathcal{R}$ , which preserves norms (see [32][Section 2.8] for more details). Hence, arbitrary relations we need to prove over  $\mathcal{R}_{kd,p}$  can be proven by showing that some corresponding relations over  $\mathcal{R}_p$  hold true.

Secondly, we observe that if we choose a proof system modulus  $q$  to be divisible by  $p$  and commit to  $(i, \mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3)$  in the “Ajtai” part of the ABDLOP commitment then the first statement in (77) is simply a system of quadratic equations in the committed messages as in Section 4. Indeed, we pick  $q = q_1 p$  where

$q_1 < p$  and then prove an equivalent quadratic relation over  $\mathcal{R}_q$ , namely:

$$q_1 [\mathbf{A}|\mathbf{B} + i\mathbf{G}|\mathbf{B}'] \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \\ \mathbf{s}_3 \end{bmatrix} = q_1 [\mathbf{A}|\mathbf{B}|\mathbf{G}|\mathbf{B}'] \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \\ i\mathbf{s}_2 \\ \mathbf{s}_3 \end{bmatrix} = q_1 \mathbf{u}. \quad (78)$$

Further, the second statement is about norms which is covered in Section 5.

Moreover, we define the identity space  $G$ . It should be designed so that we can efficiently prove that  $i \in G$  (third statement). Let  $\mathcal{B}$  be the set of non-zero binary polynomials in  $\mathcal{R}_p$ . Then, we define the identity space<sup>25</sup> as

$$G := \{i(X^k) \in \mathcal{R}'_{kd,p} : i \in \mathcal{B} \text{ and } \|i\| = \sqrt{\omega}\}.$$

We choose  $\omega$  so that the set  $G$  has size  $\approx 2^{23}$  for comparison with related work [11, 22]. Note that for appropriate  $p$ , a difference of two distinct elements from  $G$  is still invertible over  $\mathcal{R}_{kd,p}$  which is crucial for trapdoor sampling.

Note that the space  $G$  is constructed in such a way that when we map equations over  $\mathcal{R}_{kd,p}$  to  $\mathcal{R}_p^k$ , then we only need to commit to one polynomial  $i \in \mathcal{R}_p$  using our ABDLOP commitment instead of  $k$  polynomials, i.e.  $i(X^k) \in \mathcal{R}_{kd,p}$ . Similarly, we only need to send an encryption of  $i$  over  $\mathcal{R}_p$  instead of  $i(X^k)$ . Hence, for such a set  $G$ , proving  $i(X^k) \in G$  is equivalent to proving that  $i$  has binary coefficients and  $\|i\| = \sqrt{\omega}$  which is covered in Section 5.

Last but not least, we observe that including a verifiable encryption from Section 6.3 does not have a significant impact on the signature size. Indeed, identity  $i$  is already committed using the ABDLOP scheme and additionally committing to the randomness  $\mathbf{r}$  (in the ‘‘Ajtai part’’) does not increase the commitment size. Hence, the only extra cost consists of: (i) a ciphertext, (ii) masked opening of the randomness  $\mathbf{r}$ , (iii) commitments and masked openings to polynomials involved in the approximate range proof for  $\mathbf{v}$  in (75). As described in Fig. 16, for our instantiation the verifiable encryption costs 11.8KB compared to 19.5KB shown in Fig. 14.

In summary, we show in Fig. 15 how to instantiate the protocol in Fig. 10 to construct a group signature.

**Parameters.** We present our parameter selection in Fig. 16 for a group signature instantiation which achieves security level 111. We start by setting  $p = 2^{38} - 1767$  and  $q = (2^{26} - 87) \cdot p \approx 2^{64}$ . Then, we choose  $d = 128, k = 4$  and  $l = 4$ , thus  $\mathcal{R}_{kd,p} = \mathbb{Z}[X]/(X^{512} + 1)$ . Next, let  $N = 2, M = 3$  and  $\tau = 5$ , hence  $g = \lceil p^{1/5} \rceil$ . Further, we pick large enough standard deviation  $\mathfrak{s}$  used for trapdoor sampling. We know from [38] that  $\mathfrak{s} \geq 2(s_1(\mathbf{R}) + 1)\sqrt{g^2 + 1}$  where  $s_1$  is the operator norm. Note that if  $\mathbf{R}$  did not have a polynomial structure, i.e.  $R \leftarrow \{-1, 0, 1\}^{(N+M)kd \times \tau Nkd}$ , we could use upper-bounds for norms of random subgaussian matrices, e.g. [38][Lemma 2.9]. Namely, we would obtain the following bound

$$s_1(R) \leq \sqrt{(N+M)kd} + \sqrt{\tau Nkd} + 6 \approx 128$$

with probability at least  $1 - 2^{-163}$ . We found experimentally that for our structured matrix  $\mathbf{R}$  a similar bound holds with at least 99% probability

$$s_1(\mathbf{R}) \leq \psi := 113$$

and thus we set

$$\mathfrak{s} := 2(\psi + 1)\sqrt{p^{2/\tau} + 1}.$$

Further, we describe how we choose  $N$  and  $M$ , i.e. the height and the width of the matrix  $\mathbf{A}$ . Concretely, in the traceability proof, the challenger sets  $\mathbf{B} := \mathbf{A}\mathbf{R} - i^*\mathbf{G}$  and  $\mathbf{B}' = \mathbf{A}\mathbf{R}'$  where  $\mathbf{R}, \mathbf{R}' \leftarrow S_{kd,1}^{(N+M) \times \tau N}$  and  $i^* \leftarrow G$ . Additionally, it samples  $s_k^{\text{gm}} := (s_1^{\text{gm}}, s_2^{\text{gm}}, s_3^{\text{gm}}) \leftarrow D_s^{((2\tau+1)N+M)kd}$  and computes

<sup>25</sup> Previous works [16, 32] define the identity space  $G$  to be a set of integers  $\mathbb{Z}_p$  since it was easier to prove set membership  $i \in G$  with their proof system. Here, we make a small modification and set the identity space to be a subset of binary polynomials with fixed norm.

variable	description	instantiation
$\rho$	# of equations to prove	$N$
$\rho_{\text{eval}}$	# of evaluations with const. coeff. zero	2
$\nu_e$	# of exact norm proofs	2
$\nu_d$	# non-exact norm proofs	1
$\mathbf{s}_1$	committed message in the Ajtai part	$(\mathbf{s}_1^{(i)}, \mathbf{s}_2^{(i)}, \mathbf{s}_3^{(i)}, \mathbf{r}_{\text{enc}}, i)$
$\mathbf{m}$	committed message in the BDLOP part	$\emptyset$ (no message)
$f_1, \dots, f_N$	equations to prove	Equation 78
$F_1$	evaluation to prove const coeff. zero	$\sigma_{-1}(i) \left( i - \sum_{j=0}^{d-1} X^j \right)$
$F_2$	evaluation to prove const coeff. zero	$\sigma_{-1}(i) \cdot i - \omega$
$\mathbf{E}_1$	public matrix for proving $\ \mathbf{E}_1 \mathbf{s} - \mathbf{v}_1\  \leq \beta_1^{(e)}$	$[\mathbf{I}_{k(N+M+2\tau N)} \mathbf{0}]$
$\mathbf{v}_1$	public vector for proving $\ \mathbf{E}_1 \mathbf{s} - \mathbf{v}_1\  \leq \beta_1^{(e)}$	$\mathbf{0}$
$\beta_1^{(e)}$	upper-bound on $\ \mathbf{E}_1 \mathbf{s} - \mathbf{v}_1\  \leq \beta_1^{(e)}$	$\mathbf{s} \sqrt{2((2\tau + 1)N + M)kd}$
$\mathbf{E}_2$	public matrix for proving $\ \mathbf{E}_2 \mathbf{s} - \mathbf{v}_2\  \leq \beta_2^{(e)}$	$[\mathbf{0} \mathbf{0} \mathbf{0} \mathbf{I}_{K_{\text{enc}}+1}]$
$\mathbf{v}_2$	public vector for proving $\ \mathbf{E}_2 \mathbf{s} - \mathbf{v}_2\  \leq \beta_2^{(e)}$	$\mathbf{0}$
$\beta_2^{(e)}$	upper-bound on $\ \mathbf{E}_2 \mathbf{s} - \mathbf{v}_2\  \leq \beta_2^{(e)}$	$\sqrt{B_{\text{enc}}^2 + \omega}$
$\mathbf{D}_1$	public matrix for proving $\ \mathbf{D}_1 \mathbf{s} - \mathbf{u}_1\  \leq \beta_1^{(d)}$	$p_{\text{enc}}^{-1} \cdot \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{A}_{\text{enc}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{b}_{\text{enc}}^T & [\frac{p_{\text{enc}}}{2}] \end{bmatrix}$
$\mathbf{u}_1$	public vector for proving $\ \mathbf{D}_1 \mathbf{s} - \mathbf{u}_1\  \leq \beta_1^{(d)}$	$p_{\text{enc}}^{-1} \cdot \begin{bmatrix} t_0 \\ t_1 \end{bmatrix}$
$\beta_1^{(d)}$	upper-bound on $\ \mathbf{D}_1 \mathbf{s} - \mathbf{u}_1\  \leq \beta_1^{(d)}$	$B_{v,\text{enc}}$

**Fig. 15:** Instantiation of the protocol in Fig. 10 for the group signature. The variables in the first two columns refer to the ones defined in Section 5 and the ones in the last column refer to the parameters in this subsection. Variables with subscript **enc** are defined for the verifiable encryption in Section 6.3. Functions  $F_1, F_2$  are used to prove that identity  $i$  has binary coefficients and  $\|i\| = \sqrt{\omega}$ . Triples  $(\mathbf{E}_1, \mathbf{v}_1, \beta_1^{(e)})$  and  $(\mathbf{E}_2, \mathbf{v}_2, \beta_2^{(e)})$  correspond to proving exactly  $\|(\mathbf{s}_1^{(i)}, \mathbf{s}_2^{(i)}, \mathbf{s}_3^{(i)})\| \leq B$  and  $\|(\mathbf{r}_{\text{enc}}, i)\| \leq \sqrt{B_{\text{enc}}^2 + \omega}$  respectively. The last triple  $(\mathbf{D}_1, \mathbf{u}_1, \beta_1^{(d)})$  corresponds to proving approximately that  $\|\mathbf{v}_{\text{enc}}\| \leq B_{v,\text{enc}} := (B_{\text{enc}} \sqrt{K_{\text{enc}} d} / 2 + 1) \sqrt{(N_{\text{enc}} + 1)d}$  where  $\mathbf{v}_{\text{enc}}$  is defined in (75).

$\mathbf{u} := [\mathbf{A}|\mathbf{AR}|\mathbf{AR}'] sk_{i^*}^{\text{gm}}$ . It will hope that an adversary forges a signature for the identity  $i^*$ <sup>26</sup> In that case, we can extract from the forged signature the secret vector  $sk_{i^*} = (\bar{\mathbf{s}}_1, \bar{\mathbf{s}}_2, \bar{\mathbf{s}}_3)$  such that

$$[\mathbf{A}|\mathbf{AR}|\mathbf{AR}'] \begin{bmatrix} \bar{\mathbf{s}}_1 \\ \bar{\mathbf{s}}_2 \\ \bar{\mathbf{s}}_3 \end{bmatrix} = \mathbf{u} = [\mathbf{A}|\mathbf{AR}|\mathbf{AR}'] \begin{bmatrix} \mathbf{s}_1^{\text{gm}} \\ \mathbf{s}_2^{\text{gm}} \\ \mathbf{s}_3^{\text{gm}} \end{bmatrix}$$

and thus

$$\mathbf{s} := \bar{\mathbf{s}}_1 - \mathbf{s}_1^{\text{gm}} + \mathbf{R}(\bar{\mathbf{s}}_2 - \mathbf{s}_2^{\text{gm}}) + \mathbf{R}'(\bar{\mathbf{s}}_3 - \mathbf{s}_3^{\text{gm}})$$

is a MSIS solution for the matrix  $\mathbf{A}$ <sup>27</sup>. Also, with high probability we have  $\mathbf{s} \neq \mathbf{0}$  since  $sk_{i^*}^{\text{gm}}$  was chosen independently by the challenger. Now, we need to bound the norm of  $\mathbf{s}$ . In order to do so, we will use the property that for any  $\mathbf{x} \in \mathcal{R}_p^{\tau N}$ ,  $\|\mathbf{R}\mathbf{x}\| \leq s_1(\mathbf{R})\|\mathbf{x}\| \leq \psi\|\mathbf{x}\|$ . Thus, we can bound the norm of  $\mathbf{s}$  defined above using the Cauchy-Schwarz inequality as follows:

$$\begin{aligned} \|\mathbf{s}\| &\leq \|\bar{\mathbf{s}}_1 - \mathbf{s}_1^{\text{gm}}\| + \psi\|\bar{\mathbf{s}}_2 - \mathbf{s}_2^{\text{gm}}\| + \psi\|\bar{\mathbf{s}}_3 - \mathbf{s}_3^{\text{gm}}\| \\ &\leq \sqrt{1 + \psi^2 + \psi^2} \cdot \sqrt{\|\bar{\mathbf{s}}_1 - \mathbf{s}_1^{\text{gm}}\|^2 + \|\bar{\mathbf{s}}_2 - \mathbf{s}_2^{\text{gm}}\|^2 + \|\bar{\mathbf{s}}_3 - \mathbf{s}_3^{\text{gm}}\|^2}. \end{aligned}$$

<sup>26</sup> Hence, there is a  $1/|G|$  security loss.

<sup>27</sup> Since we prove the norm of  $sk_{i^*}$  exactly, there is no relaxation factor  $c$  in front of the vector  $\mathbf{u}$  as in previous works.

parameters	description	value
$p$	modulus for the group signature	$2^{38} - 1767$
$d$	ring dimension for of $\mathcal{R}$	128
$k$	$kd$ is the ring dimension of $\mathcal{R}_{kd}$	4
$N$	height of the $\mathbf{A}$ matrix	2
$M$	$N + M$ is the width of the $\mathbf{A}$ matrix	4
$\tau$	$\tau N$ is the width of the gadget matrix $\mathbf{G}$	5
$\omega$	#1's in the identity $i \in G$	3
$ G $	size of the identity space	$\approx 2^{23}$
$p_{\text{enc}}$	encryption modulus	3329
$N_{\text{enc}}$	height of $\mathbf{A}_{\text{enc}}$	4
$K_{\text{enc}}$	width of $\mathbf{A}_{\text{enc}}$	9
$\xi_{\text{enc}}$	$\xi_{\text{enc}}^K$ is the randomness distribution of $\mathbf{r}_{\text{enc}}$	$\text{Bin}_2^d$
$q$	modulus for the proof system	$\approx 2^{64}$
$l$	# factors $X^d + 1$ splits into mod $q$	4
$\gamma_1$	rejection sampling constant for $\text{cs}_1$	5
$\gamma_2$	rejection sampling constant for $\text{cs}_2$	3
$\gamma^{(e)}$	rejection sampling constant exact ARP	2
$\gamma^{(d)}$	rejection sampling constant for non-exact ARP	12
$\kappa$	maximum coefficient of a challenge in $\mathcal{C}$	2
$n$	height of matrices $\mathbf{A}_1, \mathbf{A}_2$ in ABDLOP	11
$m_1$	length of the message $\mathbf{s}_1$ in the "Ajtai" part	110
$\ell$	length of the message $\mathbf{m}$ in the "BDLOP" part	0
$\lambda$	number of garbage $g_j \in \mathcal{R}_q$ for boosting soundness	6
$m_2$	length of the randomness $\mathbf{s}_2$ in ABDLOP	41
$\nu$	randomness $\mathbf{s}_2$ is sampled from $S_\nu^{m_2}$	1
	repetition rate	27
	extra cost of adding verifiable encryption	11.8KB
	signature size	90KB
	public key size	47.5KB
	secret key size	6.3KB

**Fig. 16:** Parameter selection and concrete sizes for the group signature scheme.

Finally, we observe that we can bound the second term as:

$$\left\| \begin{bmatrix} \bar{\mathbf{s}}_1 - \mathbf{s}_1^{\text{gm}} \\ \bar{\mathbf{s}}_2 - \mathbf{s}_2^{\text{gm}} \\ \bar{\mathbf{s}}_3 - \mathbf{s}_3^{\text{gm}} \end{bmatrix} \right\|^2 \leq 2 \cdot \left( \left\| \begin{bmatrix} \bar{\mathbf{s}}_1 \\ \bar{\mathbf{s}}_2 \\ \bar{\mathbf{s}}_3 \end{bmatrix} \right\|^2 + \left\| \begin{bmatrix} \mathbf{s}_1^{\text{gm}} \\ \mathbf{s}_2^{\text{gm}} \\ \mathbf{s}_3^{\text{gm}} \end{bmatrix} \right\|^2 \right) \leq 4B^2 = (2B)^2.$$

Hence

$$\|\mathbf{s}\| \leq B_{\text{MSIS}} := 2\mathfrak{s} \cdot \sqrt{1 + 2\psi^2} \cdot \sqrt{2((2\tau + 1)N + M)kd}.$$

Thus we have to choose  $N$  such that  $\text{MSIS}_{N, N+M, B_{\text{MSIS}}}$  is hard over  $\mathcal{R}_{kd, p}$  and take into account the  $1/|G|$  security loss. Not to mention the fact that we want  $\mathbf{AR}$  to be computationally indistinguishable from a random matrix  $\mathbf{B}$ , i.e. the  $\text{MLWE}_{N, M, S_{kd, 1}}$  problem over  $\mathcal{R}_{kd, p}$  to be hard.

Parameters for the ABDLOP commitment are chosen similarly as in the previous examples. In particular, the proof system modulus  $q$  has to be large enough to prove exactly that the norm of a user secret key is at most  $B = \mathfrak{s}\sqrt{2((2\tau + 1)N + M)kd}$ . Also, we aim for repetition rate 27 similarly as in [32].

## 6.5 Product Proofs over $\mathcal{R}_p$ for a co-prime $p$

Another application of our techniques is a product proof over  $\mathcal{R}_p$  where  $p < q$  is co-prime to our proof system modulus  $q$ . Namely, suppose we want to prove  $n$  equations of the form:

$$a_i b_i = c_i \text{ for } i = 1, 2, \dots, n \quad (79)$$

where all  $a_i, b_i, c_i \in \mathcal{R}_p$ .

Note that if  $p$  was a divisor of  $q$ , i.e.  $q = kp$  for some integer  $k$ , then we would simply apply the methodology from Section 4.2 to prove  $ka_i b_i = kc_i$  over  $\mathcal{R}_q$ . This immediately implies (79).

There are two fundamental reasons why we would consider proving such statements. Firstly, this allows us to efficiently prove quadratic relations when  $p$  is small. Indeed, suppose that we choose  $p$  which is divisible by  $q$ . Recall that the soundness error of the protocols in Section 4 mainly depends on the smallest prime divisor of  $q$ , i.e.  $q_1 \leq p$ . Hence, if we wish to have small  $p$ , we would need to decrease the number of subfields  $l$  that  $\mathcal{R}_q$  splits into (so that  $p^{-d/l}$  is negligible). Moreover, if we additionally want to execute the protocol in Fig. 8, e.g. in order to prove binary or  $L_2$  norms, we would need to increase the number of garbage terms  $g_1, \dots, g_\lambda$  so that  $p^{-\lambda}$  is negligible. This, unfortunately, has a negative impact on the overall communication size.

The second reason is that, for suitable primes  $p$ , we could prove point-wise product relations  $\vec{a} \circ \vec{b} = \vec{c}$  over  $\mathbb{Z}_p$  which is a fundamental component in proving general circuit satisfiability and RICS statements [9]. Indeed, if we choose  $p$  such that  $X^d + 1$  splits into linear factors modulo  $p$ , then using Number Theory Transform identically as done in [4, 19], we reduce the problem to proving products over  $\mathcal{R}_p$ .

We first provide a naive strategy for proving (79). Namely, we commit to  $a_i, b_i, c_i$  using the ABDLOP commitment<sup>28</sup> and prove that the  $L_2$  norms of each polynomial is at most  $p\sqrt{d}/2$ . Then, we commit to each

$$k_i := \frac{a_i b_i - c_i}{p}$$

and prove that  $\|k_i\| \leq (pd + 2)\sqrt{d}/4$ . Finally, we prove quadratic equations

$$a_i b_i - c_i = p k_i \quad (80)$$

over  $\mathcal{R}_q$ .

The intuition for soundness is that if we proved that  $a_i, b_i, c_i$  and  $k_i$  have small coefficients with respect to  $q$ , and that (80) holds over  $\mathcal{R}_q$ , then this implies that  $a_i b_i - c_i = p k_i$  is true over integers since no modulo wrap-around occurs<sup>29</sup>. Consequently, we get  $a_i b_i = c_i$  over  $\mathcal{R}_p$ .

Unfortunately, the cost of this method is committing to additional  $k_i$  for each out of  $n$  equations. We circumvent this issue by not committing to  $k_i$  but instead proving that  $p^{-1}(a_i b_i - c_i) \in \mathcal{R}_q$  has small coefficients. As described before, we do that by committing to the masking polynomials  $(y_1, \dots, y_{256/d}) \in \mathcal{R}_q^{256/d}$  and computing  $(z_1, \dots, z_{256/d}) \in \mathcal{R}_q^{256/d}$  such that

$$\begin{bmatrix} \vec{z}_1 \\ \vdots \\ \vec{z}_{256/d} \end{bmatrix} := R \begin{bmatrix} p^{-1}(\overrightarrow{a_1 b_1} - \vec{c}_1) \\ \vdots \\ p^{-1}(\overrightarrow{a_n b_n} - \vec{c}_n) \end{bmatrix} + \begin{bmatrix} \vec{y}_1 \\ \vdots \\ \vec{y}_{256/d} \end{bmatrix}$$

where  $R$  is a challenge matrix and  $\overrightarrow{a_i b_i}$  is a coefficient vector of  $a_i b_i \in \mathcal{R}_q$ <sup>30</sup>. Then, we need to prove that polynomials  $z_i$  were well-formed.

<sup>28</sup> Hence, each coefficient of  $a_i, b_i, c_i$  is between  $-p/2$  and  $p/2$ .

<sup>29</sup> This strategy was already used to prove integer multiplication in [33].

<sup>30</sup> For simplicity, we omit bimodal rejection sampling which would end up having to prove cubic rather than quadratic equations.

Let us focus on the constant coefficient  $\tilde{z}_1 \in \mathbb{Z}_q$  of  $z_1$  since proving all the other ones follows identically. Then, if we denote the first row of  $R$  by  $(r_1, \dots, r_n) \in \mathcal{R}_q^n$ , we have:

$$\tilde{z}_1 = p^{-1} \sum_{i=1}^n \vec{r}_i^T (\vec{a}_i \vec{b}_i - \vec{c}_i) + [1 \ 0 \ \dots \ 0] \vec{y}_1.$$

Hence, we simply need to prove that the constant coefficient of

$$p^{-1} \sum_{i=1}^n \sigma_{-1}(r_i)(a_i b_i - c_i) + y_1 - z_1$$

is equal to zero. Note that all  $a_i, b_i, c_i$  and  $y_1$  are committed. Hence, this is a quadratic relation with an automorphism and thus we can apply the protocol in Fig. 8 to prove this property.

## 7 Working Over General Rings

Throughout the paper, we have focused on working over the polynomial ring  $\mathcal{R} = \mathbb{Z}[X]/(X^d + 1)$ , and in particular used the fact that  $s(X^{-1})$  is an automorphism in this ring. In this section, we explain how our main results can be generalized to virtually any other ring that one could be interested in. In particular, let us define  $R = \mathbb{Z}[X]/(X^d + f_{d-1}X^{d-1} + f_{d-2}X^{d-2} + \dots + f_1X \pm 1)$ , where  $f_i \in \mathbb{Z}$ .

The first thing to note is that all our protocols for proving linear a quadratic relations over  $\mathcal{R}_q$  did not use any special properties of the ring except that the challenge differences need to be invertible. For purposes of security, one should also be mindful of the “expansion factor” of the ring, which controls the growth of polynomial products in the ring – if it is too big, then the reduction from SIS becomes meaningless [30].

For our proofs over the ring  $R_q$  to be meaningfully applied to proving knowledge of inner products over  $\mathbb{Z}$ , one needs a correspondence between the inner product and the constant coefficient of a polynomial multiplication. Below, we show how one can achieve such a correspondence for any  $R$ . The multiplication of  $a \cdot b$  in the ring  $R$  can be written as a matrix-vector product  $A\vec{b}$ , where  $\vec{b}$  consists of the coefficients of  $b$  and the  $i^{\text{th}}$  column of  $A$  (if we number them starting from 0) consists of vectors whose elements are the coefficients of the polynomial  $a \cdot X^i \in R$ . It’s not hard to see that the first row of  $A$  is the vector  $\vec{a}^T \cdot M$ , where

$$M = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & \dots & 0 & \pm 1 \\ 0 & 0 & 0 & \dots & \pm 1 & c_{2,d-1} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \pm 1 & \dots & c_{d-2,d-2} & c_{d-2,d-1} \\ 0 & \pm 1 & c_{d-1,2} & \dots & c_{d-1,d-2} & c_{d-1,d-1} \end{bmatrix}, \quad (81)$$

for some integers  $c_{i,j}$  which are of no particular importance to this section. Therefore the inner product  $\langle \vec{r}, \vec{s} \rangle$  is equal to  $\vec{a}^T \cdot \vec{s}$  where  $\vec{a}^T \cdot M = \vec{r}^T$ . Since the determinant of  $M$  is  $\pm 1$ ,  $M^{-1}$  is also an integer matrix, and thus  $\vec{a}^T = \vec{r}^T \cdot M^{-1}$  is an integer vector and so  $a \in R$ .

The protocol for proving a bound on  $\|s\|^2$  over the ring  $\mathcal{R}_q$ , uses the fact that the matrix  $M^{-1}$  actually corresponds to an automorphism over  $\mathcal{R}_q$ , and so the prover does not need to create a commitment to both  $s$  and  $\vec{s}^T \cdot M^{-1}$  – the verifier can essentially derive the latter by himself. In rings where  $\vec{s}^T \cdot M^{-1}$  is not an automorphism, the prover would additionally need to commit to the polynomial corresponding to  $\vec{r} = \vec{s}^T \cdot M^{-1}$ , and then give a linear proof showing that this relationship is indeed satisfied, along with the proof on the bound of  $\|s\|^2 = \vec{r} \cdot s$ . The modification for proving that  $s$  contains only 0/1 coefficients would proceed in the same manner. Proving component-wise products over general rings  $R$  can also be done, but ends up again doubling the committed vector. Recall that the idea when working over the ring  $\mathcal{R}_q$  was to pick a prime  $p \ll q$  such that  $X^d + 1$  fully splits modulo  $p$  and then embed the coefficients into the CRT slots. If, for a particular ring  $R$ , there is no such  $p$ , then one would need to use a different ring than the one used for the commitment scheme which does have such a  $p$ , and make sure that multiplication of committed

values over this ring corresponds to the one used in the commitment scheme. One way to do this is to only commit to polynomials of less than half the degree of the ring, so that multiplications in both rings is the same as over  $\mathbb{Z}[X]$ .

## References

1. S. Agrawal, D. Boneh, and X. Boyen. Efficient lattice (H)IBE in the standard model. In *EUROCRYPT*, pages 553–572, 2010.
2. M. Ajtai. Generating hard instances of lattice problems (extended abstract). In *STOC*, pages 99–108, 1996.
3. T. Attema, R. Cramer, and L. Kohl. A compressed  $\mathbb{Z}$ -protocol theory for lattices. In *CRYPTO (2)*, volume 12826 of *Lecture Notes in Computer Science*, pages 549–579. Springer, 2021.
4. T. Attema, V. Lyubashevsky, and G. Seiler. Practical product proofs for lattice commitments. In *CRYPTO (2)*, volume 12171 of *Lecture Notes in Computer Science*, pages 470–499. Springer, 2020.
5. W. Banaszczyk. New bounds in some transference theorems in the geometry of numbers. *Mathematische Annalen*, 296(1):625–635, Dec 1993.
6. C. Baum, I. Damgård, V. Lyubashevsky, S. Oechsner, and C. Peikert. More efficient commitments from structured lattice assumptions. In *SCN*, pages 368–385, 2018.
7. C. Baum and V. Lyubashevsky. Simple amortized proofs of shortness for linear relations over polynomial rings. *IACR Cryptology ePrint Archive*, 2017:759, 2017.
8. C. Baum and A. Nof. Concretely-efficient zero-knowledge arguments for arithmetic circuits and their application to lattice-based cryptography. In *Public Key Cryptography (1)*, pages 495–526. Springer, 2020.
9. E. Ben-Sasson, A. Chiesa, M. Riabzev, N. Spooner, M. Virza, and N. P. Ward. Aurora: Transparent succinct arguments for R1CS. In *EUROCRYPT (1)*, volume 11476 of *Lecture Notes in Computer Science*, pages 103–128. Springer, 2019.
10. W. Beullens. Sigma protocols for mq, PKP and sis, and fishy signature schemes. In *EUROCRYPT (3)*, volume 12107 of *Lecture Notes in Computer Science*, pages 183–211. Springer, 2020.
11. W. Beullens, S. Dobson, S. Katsumata, Y. Lai, and F. Pintore. Group signatures and more from isogenies and lattices: Generic, simple, and efficient. *IACR Cryptol. ePrint Arch.*, page 1366, 2021.
12. J. Bootle, V. Lyubashevsky, and G. Seiler. Algebraic techniques for short(er) exact lattice-based zero-knowledge proofs. In *CRYPTO (1)*, volume 11692 of *Lecture Notes in Computer Science*, pages 176–202. Springer, 2019.
13. J. W. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. M. Schanck, P. Schwabe, G. Seiler, and D. Stehlé. CRYSTALS - kyber: A cca-secure module-lattice-based KEM. In *2018 IEEE European Symposium on Security and Privacy, EuroS&P*, pages 353–367, 2018.
14. R. Canetti, Y. Lindell, R. Ostrovsky, and A. Sahai. Universally composable two-party and multi-party secure computation. In *STOC*, pages 494–503. ACM, 2002.
15. J. D’Anvers, A. Karmakar, S. S. Roy, and F. Vercauteren. Saber: Module-lwr based key exchange, cpa-secure encryption and cca-secure KEM. In *AFRICACRYPT*, pages 282–305, 2018.
16. R. del Pino, V. Lyubashevsky, and G. Seiler. Lattice-based group signatures and zero-knowledge proofs of automorphism stability. In *ACM Conference on Computer and Communications Security*, pages 574–591. ACM, 2018.
17. L. Ducas, A. Durmus, T. Lepoint, and V. Lyubashevsky. Lattice signatures and bimodal gaussians. In *CRYPTO (1)*, pages 40–56, 2013.
18. L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler, and D. Stehlé. Crystals-dilithium: A lattice-based digital signature scheme. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2018(1):238–268, 2018.
19. M. F. Esgin, N. K. Nguyen, and G. Seiler. Practical exact proofs from lattices: New techniques to exploit fully-splitting rings. In *ASIACRYPT (2)*, pages 259–288, 2020.
20. M. F. Esgin, R. Steinfeld, J. K. Liu, and D. Liu. Lattice-based zero-knowledge proofs: New techniques for shorter and faster constructions and applications. In *CRYPTO (1)*, pages 115–146. Springer, 2019.
21. M. F. Esgin, R. Steinfeld, and R. K. Zhao. Matric+: More efficient post-quantum private blockchain payments. *IACR Cryptol. ePrint Arch.*, page 545, 2021.
22. M. F. Esgin, R. K. Zhao, R. Steinfeld, J. K. Liu, and D. Liu. Matric: Efficient, scalable and post-quantum blockchain confidential transactions protocol. In *CCS*, pages 567–584. ACM, 2019.
23. C. Gentry, S. Halevi, and V. Lyubashevsky. Practical non-interactive publicly verifiable secret sharing with thousands of parties. *IACR Cryptol. ePrint Arch.*, page 1397, 2021.
24. J. Hoffstein, J. Pipher, and J. H. Silverman. NTRU: A ring-based public key cryptosystem. In *ANTS*, pages 267–288, 1998.

25. A. Langlois and D. Stehlé. Worst-case to average-case reductions for module lattices. *Designs, Codes and Cryptography*, 75(3):565–599, 2015.
26. 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 (2)*, pages 1–31. Springer, 2016.
27. S. Ling, K. Nguyen, D. Stehlé, and H. Wang. Improved zero-knowledge proofs of knowledge for the ISIS problem, and applications. In *PKC*, pages 107–124, 2013.
28. V. Lyubashevsky. Fiat-Shamir with aborts: Applications to lattice and factoring-based signatures. In *ASIACRYPT*, pages 598–616, 2009.
29. V. Lyubashevsky. Lattice signatures without trapdoors. In *EUROCRYPT*, pages 738–755, 2012.
30. V. Lyubashevsky and D. Micciancio. Generalized compact knapsacks are collision resistant. In *ICALP (2)*, pages 144–155, 2006.
31. V. Lyubashevsky and G. Neven. One-shot verifiable encryption from lattices. In *EUROCRYPT*, 2017.
32. V. Lyubashevsky, N. K. Nguyen, M. Plançon, and G. Seiler. Shorter lattice-based group signatures via "almost free" encryption and other optimizations. In *ASIACRYPT (4)*, pages 218–248. Springer, 2021.
33. V. Lyubashevsky, N. K. Nguyen, and G. Seiler. Practical lattice-based zero-knowledge proofs for integer relations. In *CCS*, pages 1051–1070. ACM, 2020.
34. V. Lyubashevsky, N. K. Nguyen, and G. Seiler. Shorter lattice-based zero-knowledge proofs via one-time commitments. In *Public Key Cryptography (1)*, pages 215–241. Springer, 2021.
35. V. Lyubashevsky, N. K. Nguyen, and G. Seiler. SMILE: set membership from ideal lattices with applications to ring signatures and confidential transactions. In *CRYPTO (2)*, volume 12826 of *Lecture Notes in Computer Science*, pages 611–640. Springer, 2021.
36. V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. In *EUROCRYPT*, pages 1–23, 2010.
37. V. Lyubashevsky and G. Seiler. Short, invertible elements in partially splitting cyclotomic rings and applications to lattice-based zero-knowledge proofs. In *EUROCRYPT (1)*, pages 204–224. Springer, 2018.
38. D. Micciancio and C. Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In *EUROCRYPT*, pages 700–718, 2012.
39. O. Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6), 2009.
40. C. Schnorr. Efficient identification and signatures for smart cards. In *CRYPTO*, pages 239–252, 1989.
41. J. Stern. A new identification scheme based on syndrome decoding. In *CRYPTO*, pages 13–21, 1993.
42. R. Yang, M. H. Au, Z. Zhang, Q. Xu, Z. Yu, and W. Whyte. Efficient lattice-based zero-knowledge arguments with standard soundness: Construction and applications. In *CRYPTO (1)*, pages 147–175. Springer, 2019.