

國立臺灣大學理學院數學所

碩士論文

Department of Mathematics

College of Science

National Taiwan University

Master's Thesis



格基底整數關係的承諾與證明

Lattice-Based Commit-and-Prove Proofs for Integer  
Relations

胡政賢

Cheng-Hsien Hu

指導教授：陳君明 博士

Advisor: Jiun-Ming Chen Ph.D.

中華民國 113 年 2 月

February 2024

# Acknowledgements



My deepest gratitude goes to my advisor, Ph.D. Jiun-Ming Chen, for his invaluable assistance and guidance. He motivated and inspired me to have the right attitude as a researcher. He provided informative instruction and critically read the thesis.

# 摘要



在 [LNS20] 中，作者設計兩個整數關係的晶格基底零知識證明協定，分別是證明第三個秘密整數是另外兩個秘密整數的和，而另一個則是其乘法版本的，然而這兩個協定都要求底層的環擁有多個CRT slots，這導致了無法忽視的可靠度誤差。

依據 [LNP22] 的基礎，我們建構了兩個零知識協定，用於證明先前所提及的整數問題，而無需對底層的環進行先前的限制。此外，我們將加法版本協定推廣到證明 $k$ 個整數之和，其中 $k$ 取決於秘密整數的二進位表示。

**關鍵字:** 整數關係的晶格基底零知識證明協定、ABDLOP承諾計畫、承諾與證明協定、MSIS問題、Extended-MLWE問題

# Abstract



In [LNS20], the authors designed two zero-knowledge protocols for integer relations. The underlying rings of the two lattice-based protocols possess many CRT slots, which has a negative effect on soundness error. One is for proving that the third secret integer is the sum of two other secret integers, while the other is the multiplicative version. Based on the foundation laid by [LNP22], we construct two zero-knowledge protocols dealing with the original problem without the previous requirement for the underlying ring. Moreover, we generalize the addition protocol from sum of two integers to sum of  $k$  integers, dependent of bits representing our secret ones.

**Keywords:** Lattice-based zero-knowledge protocol for integer relations, ABDLOP commitment scheme, Commit-and-prove protocol, MSIS, Extended-MLWE

# Contents



## Acknowledgements

### 摘要

**i**  
**ii**

## Abstract

**iii**

## 1 Introduction

**1**

## 2 Preliminaries

**3**

2.1 Notation . . . . .	3
2.2 Cyclotomic Rings . . . . .	4
2.3 Discrete Gaussian Distributions . . . . .	5
2.4 Module-SIS and Module-LWE Problems . . . . .	5
2.5 Rejection Sampling . . . . .	7
2.6 Challenge Space . . . . .	8

## 3 Proofs for Quadratic Relations

**10**

3.1 Commit-and-prove Protocol . . . . .	10
3.2 Main Protocol . . . . .	12

## 4 Applications to Integer Relation

**20**

4.1 Integer Addition . . . . .	20
4.2 Integer Multiplication . . . . .	28

## References

**30**

# List of Figures



1	Two rejection sampling algorithms: The sampling $\text{Rej}_1$ in [Lyu12] and the other $\text{Rej}_2$ in [LNS21]. . . . .	8
2	Commit-and-prove protocol $\Pi((\mathbf{s}_2, \mathbf{s}_1, \mathbf{m}), \sigma, (F_i)_{i=1}^N, (f_j)_{j=1}^M)$ satisfying (i) $\ \bar{c}\mathbf{s}_i\  \leq 2s_i\sqrt{2m_id}$ (ii) $\mathbf{t}_A = \mathbf{A}_1\mathbf{s}_1 + \mathbf{A}_2\mathbf{s}_2, \mathbf{t}_B = \mathbf{B}\mathbf{s}_2 + \mathbf{m}$ (iii) $F_j(\sigma^i(\mathbf{s}_1), \sigma^i(\mathbf{m})) = 0$ for all $j \leq N$ and (iv) $\tilde{f}_j(\sigma^i(\mathbf{s}_1), \sigma^i(\mathbf{m})) = 0$ for all $j \leq M$ where $\mathbf{R}_2, \mathbf{r}_1, r_0$ is defined as $F(\mathbf{x}) = \mathbf{x}^t\mathbf{R}_2\mathbf{x} + \mathbf{r}_1^t\mathbf{x} + r_0$ . . . . .	14
3	Verification equation for Figure 2 . . . . .	15
4	Proof of general integer addition where $2k < d$ . . . . .	25
5	Proof of integer multiplication where $N \leq d/2$ . . . . .	29

# 1 Introduction

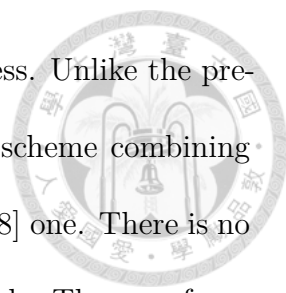


For lattice-based cryptography, one of the fundamental hardness assumptions is that finding a low-norm  $\mathbf{s}$  such that

$$\mathbf{As} = \mathbf{t} \pmod{q} \tag{I}$$

is computationally hard. In earlier times, Stern [Ste94] used a combinatorial algorithm for exactly proving (I) to prove that  $\|\mathbf{s}\|_\infty$  is bounded via exposing a random permutation of  $\mathbf{s}$ . However, the soundness of those protocols exploiting this technique was  $2/3$ , and thus, they had to repeat about 200 times to reach a negligible soundness error. Repeating the protocol leads to more than 1 megabyte (e.g. [LNSW13]) in proof size.

Another more algebraic method for exactly proving (I) is to make use of lattice-based commitments and zero-knowledge proofs about committed values to show the relation between the coefficients of  $\mathbf{s}$  and a bound of  $\|\mathbf{s}\|_\infty$ . The advent of such kind of protocols (e.g. [BLS19, ESLL19]) reduced the proof size to the order of a number of hundred kilobytes. The recent work of [ALS20, ENS20] created efficient zero-knowledge proof systems for proving products of polynomials over a ring and linear relationships among the CRT coefficients of committed values. Subsequently, [LNS20] building on [ALS20, ENS20] developed efficient zero-knowledge proofs peculiarly designed for integer addition and multiplication respectively. Yet, this approach has a potential factor to deter its efficiency. An incompatibility arises due to the simultaneous need for the underlying ring to possess numerous CRT slots and the requirement for the protocol to have a negligible soundness error. Therefore,



a segment of the protocol must be reiterated to enhance soundness. Unlike the previous approach, [LNP22] proposed a lattice-based commitment scheme combining the Ajtai [Ajt96] commitment scheme as well as BDLOP [BDL<sup>+</sup>18] one. There is no requirement for this protocol to recommit to  $\mathbf{s}$  in Chinese Remainder Theorem form; hence we do not have a requirement having an impact on the soundness. Moreover, not needing to commit  $\mathbf{s}$  in the BDLOP way cuts down on the proof size.

In this paper, our main result is to design an efficient zero-knowledge protocol building on [LNS20, LNP22] for arbitrary sums of integer addition and multiplication without requiring our underlying ring to have a lot of CRT slots, which theoretically claimed to reduce the proof size and improve the efficiency of the protocol. On the other hand, we reduce the number of polynomials to prove that a given element in  $\mathcal{R}_q$  is indeed an integer dramatically by applying lemma 4.1.2, which improves the efficiency for the prover.

In many real-world scenarios, both protocols for proving integer relations are helpful. For instance, consider an online auction where all the auction participants do not want to expose their own fortune in their accounts. The bidder wants to purchase  $n_i$  units of item  $i$  at a price of  $p_i$ . Since the final bidding price is dependent on the bids of those who did not win, getting rid of invalid bids to prevent unnecessary price inflation is imperative. Thus, the participants should prove that their accounts have more property than  $\sum_i n_i \cdot p_i$ . In this situation, both of our protocols that yield rather short and efficient proofs can be utilized.



## 2 Preliminaries



### 2.1 Notation

Let  $q = q_1 q_2 \dots q_n$ , the product of distinct  $n$  odd primes where  $q_1 < q_2 < \dots < q_n$ . And let  $\mathbb{Z}_q$  be the ring of the rational integers modulo  $q$  defined above. We write  $\vec{v} \in \mathbb{Z}^k$  and  $\vec{w} \in \mathbb{Z}_q^k$  to represent a vector over  $\mathbb{Z}$  and a vector over the ring  $\mathbb{Z}_q$  respectively. We utilize regular capital letters to denote matrices over  $\mathbb{Z}$  or  $\mathbb{Z}_q$ . By convention, vectors are considered as column vectors. Given two vectors  $\vec{u}, \vec{v} \in \mathbb{R}^n$ ,  $\vec{u} \parallel \vec{v}$  denotes the usual concatenation of  $\vec{u}$  and  $\vec{v}$ . Given a distribution  $D$ ,  $z \leftarrow D$  denotes that  $z$  is sampled from  $D$ . Similarly, given a set  $S$ ,  $x \leftarrow S$  represents  $x$  is sampled uniformly from the set  $S$ .  $[n]$  denotes the set  $\{1, \dots, n\}$ .

Let  $d$  be a power of 2 and  $p$  be a natural number. We define  $\mathcal{R}$  and  $\mathcal{R}_p$  as  $\mathbb{Z}[X]/(X^d + 1)$  and  $\mathbb{Z}_p[X]/(X^d + 1)$ , respectively. In this paper, we employ lowercase letters to signify elements in  $\mathcal{R}$  or  $\mathcal{R}_p$ , while bold lowercase letters are employed to indicate column vectors whose components lie in  $\mathcal{R}$  or  $\mathcal{R}_p$ . Correspondingly, bold uppercase letters are utilized to represent matrices with entries within these rings. When considering a polynomial  $f \in \mathcal{R}$  (or  $\mathcal{R}_p$ ), the vector  $\vec{f} \in \mathbb{Z}$  (or  $\mathbb{Z}_p$ ) denotes the coefficient vector. As a convention, we often deem  $f \in \mathcal{R}_p$  as a polynomial of degree at most  $d - 1$  in  $\mathbb{Z}_p$ . Consequently, we can express the  $i^{\text{th}}$  coefficient of the polynomial as  $f_i \in \mathbb{Z}_p$ . For convenience, we sometimes denote  $f_0$  by  $\tilde{f}$ . We define the inner product in  $\mathcal{R}^k$ . For  $u, v \in \mathcal{R}$ , the notation  $\langle u, v \rangle$  denotes  $\sum_{i=0}^{d-1} u_i \cdot v_i \in \mathbb{Z}$ . This inner product can be naturally extended to  $\mathcal{R}^k$ .

For  $w \in \mathbb{Z}_q$ ,  $\|w\|_\infty$  denotes the absolute value in  $\mathbb{R}$  of the unique representative  $r$  of  $w$  where  $w \equiv r \pmod{q}$  and  $[-\frac{q}{2}] \leq r < [\frac{q}{2}]$ . Then, we can define  $\ell_\infty$  and  $\ell_p$  norms

for element  $w = w_0 + w_1X + \cdots + w_{d-1}X^{d-1} \in \mathcal{R}_q$  as below:

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



It is nature to extend this norm from  $\mathcal{R}_q$  to  $\mathbf{w} = (w_1, \dots, w_k) \in \mathcal{R}_q^k$  via defining

$$\|\mathbf{w}\|_\infty = \max_j \|w_j\|_\infty, \quad \|\mathbf{w}\|_2 = \sqrt{\|w_1\|_2^2 + \cdots + \|w_k\|_2^2}.$$

$\|\mathbf{w}\| := \|\mathbf{w}\|_2$  by default. We represent  $S_\gamma = \{f \in \mathcal{R}_q : \|f\|_\infty \leq \gamma\}$ .

## 2.2 Cyclotomic Rings

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

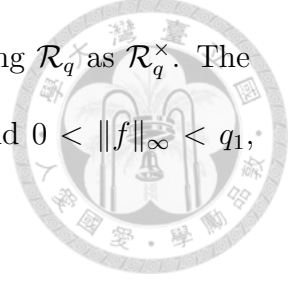
$$\mathbb{Z}_{2d}^\times \xrightarrow{\sim} \text{Aut}(\mathcal{R}) : i \mapsto \sigma_i,$$

where  $\sigma_i : X \mapsto X^i$ . Assume that each prime factor  $q_i$  factorizes into 2 prime ideals of degree  $\frac{d}{2}$  within the ring  $\mathcal{R}$ . That is,  $X^d + 1 = \phi_1\phi_2 \pmod{q_i}$  with irreducible polynomials  $\phi_j$  of degree  $\frac{d}{2}$ . Moreover, we assume that 4 is the highest order of power of 2 in  $\mathbb{Z}_{q_i}$ , in other words,  $q_i - 1 \equiv 4 \pmod{8}$ . Hence, we have

$$X^d + 1 \equiv (X^{\frac{d}{2}} - \zeta_i)(X^{\frac{d}{2}} - \zeta_i^3) \pmod{q_i}.$$

We utilize invertible criterion below following from [LNP22, Lemma 2.6].

**Lemma 2.2.1.** *Let  $p$ , an odd prime, be congruent to 5 modulo 8. Consider any  $c \in \mathcal{R}_p$  satisfying  $\sigma_{-1}(c) = c$ . Then,  $c$  is invertible in  $\mathcal{R}_p$  if and only if  $c \neq 0$ .*



By convention, we denote the set of invertible elements in ring  $\mathcal{R}_q$  as  $\mathcal{R}_q^\times$ . The lemma above asserts that for a given  $f \in \mathcal{R}_q$ , if  $\sigma_{-1}(f) = f$  and  $0 < \|f\|_\infty < q_1$ , then  $f \in \mathcal{R}_q^\times$ .

## 2.3 Discrete Gaussian Distributions

We recall the discrete Gaussian distribution employed in the rejection sampling.

**Definition 2.3.1.** We define the discrete Gaussian distribution over  $\mathcal{R}^k$  centered at  $\mathbf{c} \in \mathcal{R}^k$  with standard deviation  $\mathfrak{s} > 0$  as

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

By default, we write  $D_{\mathfrak{s}}^k$  to represent the distribution centered at  $\mathbf{0} \in \mathcal{R}^k$ .

Next, we introduce the lemma derived from [Ban93, Lemma 1.5(i)].

**Lemma 2.3.2.** *Let  $\mathbf{z} \leftarrow D_{\mathfrak{s}}^m$ . Then  $\Pr[\|\mathbf{z}\| > t \cdot \mathfrak{s}\sqrt{md}] < (te^{\frac{1-t^2}{2}})^{md}$ .*

## 2.4 Module-SIS and Module-LWE Problems

**Definition 2.4.1 (MSIS $_{\kappa, m, \beta}$ ).** Given positive integers  $\kappa, m$  and  $0 < \beta < q$ . For a given  $\mathbf{A} \leftarrow \mathcal{R}_q^{\kappa \times m}$ , the Module-SIS problem, characterized by parameters  $\kappa, m$ , and  $\beta$ , involves the search for a vector  $\mathbf{x} \in \mathcal{R}_q^m$  such that  $\mathbf{A}\mathbf{x} = \mathbf{0}$  and  $0 < \|\mathbf{x}\| \leq \beta$ . We say that a probabilistic polynomial time adversary  $\mathcal{A}$  has advantage  $\epsilon$  in solving M-SIS $_{\kappa, m, \beta}$  if

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

**Definition 2.4.2** ( $\text{MLWE}_{m,\lambda,\chi}$ ). Let  $m$  and  $\lambda$  be positive integers and  $\chi$  be an error distribution over  $\mathcal{R}_q^m$ . The Module-LWE problem with parameters  $m, \lambda$  and distribution  $\chi$  asks to distinguish  $(\mathbf{A}, \mathbf{b}) \leftarrow \mathcal{R}_q^{m \times \lambda} \times \mathcal{R}_q^m$  from  $(\mathbf{A}, \mathbf{As} + \mathbf{e})$  with  $\mathbf{A} \leftarrow \mathcal{R}_q^{m \times \lambda}$ , secret vector  $\mathbf{s} \leftarrow \chi^\lambda$ , and  $\mathbf{e} \leftarrow \chi^m$ . We say that a probabilistic polynomial-time adversary  $\mathcal{A}$  has advantage  $\epsilon$  in solving  $\text{MLWE}_{m,\lambda,\chi}$  if

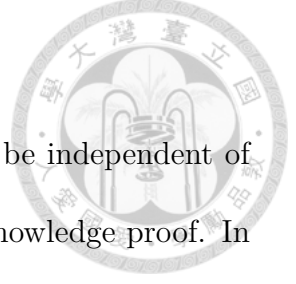
$$|\Pr[1 \leftarrow \mathcal{A}((\mathbf{A}, \mathbf{b})) \mid \mathbf{A} \leftarrow \mathcal{R}_q^{m \times \lambda}; \mathbf{b} \leftarrow \mathcal{R}_q^m] - \Pr[1 \leftarrow \mathcal{A}((\mathbf{A}, \mathbf{As} + \mathbf{e})) \mid \mathbf{A} \leftarrow \mathcal{R}_q^{m \times \lambda}; \mathbf{s} \leftarrow \chi^\lambda; \mathbf{e} \leftarrow \chi^m]| \geq \epsilon.$$

Now, we introduce a variant of the MLWE problem while using the same notation as [LNP22].

**Definition 2.4.3** ((simplified) Extended-  $\text{MLWE}_{m,\lambda,\chi,\mathcal{C},\mathfrak{s}}$ ). The Extended-Module-LWE problem with parameters  $m, \lambda \in \mathbb{N}$ , distribution  $\chi$  over  $\mathcal{R}_q$ , challenge space  $\mathcal{C} \subseteq \mathcal{R}_q$ , and standard deviation  $\mathfrak{s}$  for discrete Gaussian distribution centered at 0 asks to distinguish  $(\mathbf{A}, \mathbf{u}, c, \mathbf{z}, \text{sign}(\langle \mathbf{z}, \mathbf{c}\mathbf{r} \rangle))$  for  $\mathbf{A} \leftarrow \mathcal{R}_q^{m \times (m+\lambda)}$ ,  $\mathbf{u} \leftarrow \mathcal{R}_q^m$ ,  $c \leftarrow \mathcal{C}$ , and  $\mathbf{z} \leftarrow D_{\mathfrak{s}}^{(m+\lambda)}$  from  $(\mathbf{A}, \mathbf{Ar}, c, \mathbf{z}, \text{sign}(\langle \mathbf{z}, \mathbf{c}\mathbf{r} \rangle))$  for  $\mathbf{A} \leftarrow \mathcal{R}_q^{m \times (m+\lambda)}$ , secret vector  $\mathbf{r} \leftarrow \chi^{m+\lambda}$ ,  $c \leftarrow \mathcal{C}$ , and  $\mathbf{z} \leftarrow D_{\mathfrak{s}}^{(m+\lambda)}$ , where  $\text{sign}(b) = 0$  if  $b < 0$  and 1 otherwise. Thus, we say that a probabilistic polynomial-time adversary  $\mathcal{A}$  has advantage  $\epsilon$  in solving Extended- $\text{MLWE}_{m,\lambda,\chi,\mathcal{C},\mathfrak{s}}$  if

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

with  $\mathbf{s} = \text{sign}(\langle \mathbf{z}, \mathbf{c}\mathbf{r} \rangle)$

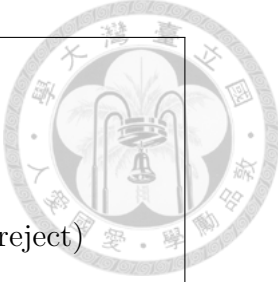


## 2.5 Rejection Sampling

The prover outputs a vector  $\mathbf{z}$  with a distribution that must be independent of the secret information or randomness  $\mathbf{r}$  in a lattice-based zero-knowledge proof. In this protocol,  $\mathbf{z}$  is computed as  $\mathbf{y} + c\mathbf{r}$ , where  $\mathbf{y}$  is a "masking" vector,  $c \leftarrow \mathcal{C}$  denotes a challenge polynomial, and  $\mathbf{r}$  represents the secret vector or randomness employed for prover's commitments. To eliminate the dependence of  $\mathbf{z}$  on  $\mathbf{r}$ , we employ rejection sampling.

**Lemma 2.5.1** (Rejection Sampling [Lyu12],[LNS21]). *Let  $W \subseteq \mathcal{R}^m$  be a set of polynomials with the norm not exceeding  $N$ ,  $\chi$  be a probability distribution over  $W$ , and a fixed standard deviation  $\mathfrak{s} = \gamma \cdot N$ . The following two statements are true.*

1. *Let  $M = e^{14/\gamma+1/(2\gamma^2)}$ . Sample  $\mathbf{v} \leftarrow \chi$  and  $\mathbf{y} \leftarrow D_{\mathfrak{s}}^m$ , compute  $\mathbf{z} = \mathbf{y} + \mathbf{v}$  and run  $b \leftarrow \text{Rej}_1(\mathbf{z}, \mathbf{v}, \mathfrak{s})$  as defined in Figure 1. Then*
  - (a)  $\Pr[b = 0] \geq (1 - 2^{-128})/M$ .
  - (b)  $\Delta(\chi \times D_{\mathfrak{s}}^m, \mathcal{F}) \leq 2^{-128}$  where  $\mathcal{F}$  is the probability distribution for  $(\mathbf{v}, \mathbf{z})$  conditioned on  $b = 0$ .
2. *Let  $M = e^{1/(2\gamma^2)}$ . Sample  $\mathbf{v} \leftarrow \chi$  and  $\mathbf{y} \leftarrow D_{\mathfrak{s}}^m$ , compute  $\mathbf{z} = \mathbf{y} + \mathbf{v}$  and run  $b \leftarrow \text{Rej}_2(\mathbf{z}, \mathbf{v}, \mathfrak{s})$  as defined in Figure 1.*
  - (a)  $\Pr[b = 0] \geq 1/(2M)$ .
  - (b) *The distribution for  $(\mathbf{v}, \mathbf{z})$  conditioned on  $b = 0$  is indeed the probability distribution  $\mathcal{P}$ .  $\mathcal{P}$  is set by sampling  $\mathbf{v} \leftarrow \chi$  as well as  $\mathbf{z} \leftarrow D_{\mathfrak{s}}^m$  that are conditioned on  $\langle \mathbf{v}, \mathbf{z} \rangle \geq 0$ , and then outputting  $(\mathbf{v}, \mathbf{z})$ .*



<u>Rej<sub>1</sub>(<math>\mathbf{z}, \mathbf{v}, \mathfrak{s}</math>)</u>	<u>Rej<sub>2</sub>(<math>\mathbf{z}, \mathbf{v}, \mathfrak{s}</math>)</u>
1: $u \leftarrow [0, 1)$	1: If $\langle \mathbf{z}, \mathbf{v} \rangle < 0$
2: If $u > \frac{1}{M} \cdot e^{\frac{-2\langle \mathbf{z}, \mathbf{v} \rangle + \ \mathbf{v}\ ^2}{2\mathfrak{s}^2}}$	2: return 1 (i.e. reject)
3: return 1 (i.e. reject)	3: $u \leftarrow [0, 1)$
4: Else	4: If $u > \frac{1}{M} \cdot e^{\frac{-2\langle \mathbf{z}, \mathbf{v} \rangle + \ \mathbf{v}\ ^2}{2\mathfrak{s}^2}}$
5: return 0 (i.e. accept)	5: return 1 (i.e. reject)
	6: Else
	7: return 0 (i.e. accept)

Figure 1: Two rejection sampling algorithms: The sampling  $\text{Rej}_1$  in [Lyu12] and the other  $\text{Rej}_2$  in [LNS21].

Consider how the parameters  $\mathfrak{s}$  and  $M$  are selected in the preceding lemma. To be more precise, the repetition rate  $M$  is selected to serve as an upper bound for:

$$\frac{D_{\mathfrak{s}}^m(\mathbf{x})}{D_{\mathbf{v}, \mathfrak{s}}^m(\mathbf{x})} = \exp\left(\frac{-2\langle \mathbf{x}, \mathbf{v} \rangle + \|\mathbf{v}\|^2}{2\mathfrak{s}^2}\right) \leq \exp\left(\frac{28\|\mathbf{v}\| + \|\mathbf{v}\|^2}{2\mathfrak{s}^2}\right).$$

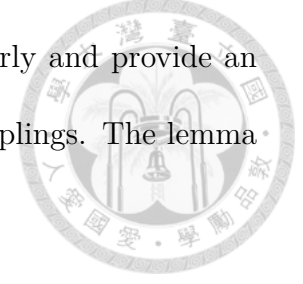
Note that we use the fact following from [Ban93],[Lyu12] that  $|\langle \mathbf{z}, \mathbf{v} \rangle| < 14\|\mathbf{v}\|$  for  $\mathbf{z} \leftarrow D_{\mathfrak{s}}^m$ . For rejection sampling  $\text{Rej}_2(\mathbf{z}, \mathbf{v}, \mathfrak{s})$ , it requires  $\mathbf{z}$  to satisfy  $\langle \mathbf{z}, \mathbf{v} \rangle \leq 0$ , or the protocol aborts. we can set  $M$  with this restriction as follows:

$$M := \exp\left(\frac{\|\mathbf{v}\|^2}{2\mathfrak{s}^2}\right).$$

## 2.6 Challenge Space

The input of the rejection sampling used in protocol should be  $\mathbf{y} \leftarrow D_{\mathfrak{s}}^{\ell}$  and  $\mathbf{v} = c\mathbf{r}$  where challenge  $c \in \mathcal{R}$  and secret  $\mathbf{r} \in \mathcal{R}_q^{\ell}$  with notation defined in previous

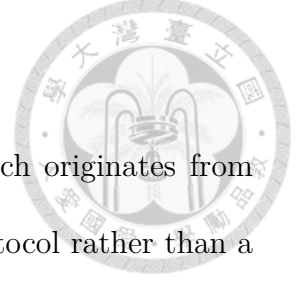
subsection. Thus, we need to choose the challenge space properly and provide an upper bound for  $\|\mathbf{c}\mathbf{r}\|$  to set our deviation for two rejection samplings. The lemma below following from [LNP22] gives a bound for  $\mathbf{c}\mathbf{r}$ .



**Lemma 2.6.1.** *Let  $\mathbf{r} \in \mathcal{R}^m$  and  $c \in \mathcal{R}$ . Fix any  $k$ , power of 2, we obtain  $\|\mathbf{c}\mathbf{r}\| \leq \sqrt[2k]{\|\sigma_{-1}(c^k)c^k\|_1} \cdot \|\mathbf{r}\|$ .*

Given a power-of-two  $k$ , we define the challenge space  $\mathcal{C}$ , by this lemma, as the set  $\{c \in S_\kappa : \sqrt[2k]{\|\sigma_{-1}(c^k)c^k\|_1} \leq \eta \wedge \sigma_{-1}(c) = c\}$ . To ensure the invertibility of the difference between two challenges, we need  $\kappa < q_1/2$ , which follows from Lemma 2.2.1. Furthermore, for attaining negligible error soundness relying on the MSIS assumption, the cardinality of the challenge space  $|\mathcal{C}|$  should be exponentially large.

### 3 Proofs for Quadratic Relations



Before we start to introduce the protocol in this paper which originates from [LNP22, Fig. 8], we say this protocol is a commit-and-prove protocol rather than a zero knowledge protocol. Hence, we need to introduce the definition of a commit-and-prove protocol.

#### 3.1 Commit-and-prove Protocol

**Definition 3.1.1.** Let  $R_L$  be a polynomial-time verifiable relation consisting of  $(ck, x, w)$ . We call  $ck$  the commitment key,  $x$  the statement, and  $w$  the witness.

1. The language  $L_{ck}$  is a set where statement  $x \in L_{ck}$  if there is a witness  $w$  such that  $(ck, x, w) \in R_L$ .
2. Let  $\lambda$  be the security parameter. A Commit-and-prove protocol is a quadruple of algorithms  $(Gen, Com, Prove, Verify)$ . We require that  $Gen$  and  $Com$  are deterministic and that the other two algorithms are probabilistic.
  - $Gen(1^\lambda)$  : It outputs a commitment key  $ck$ . The commitment key  $ck$  identifies a randomness space  $\mathcal{R}_{ck}$ , a message space  $\mathcal{M}_{ck}$ , and a commitment space  $\mathcal{C}_{ck}$ .
  - $Com_{ck}(m, r)$  :  $Com$  takes the commitment key  $ck$ , a message  $m$ , as well as a randomness  $r$  as input and outputs a commitment  $c \in \mathcal{C}_{ck}$ .
  - $Prove_{ck}(x, ((m_1, r_1), \dots, (m_n, r_n)))$  :  $Prove$  takes as input a commitment key  $ck$ , a statement  $x$  and commitment openings  $m_i \in \mathcal{M}_{ck}, r_i \in \mathcal{R}_{ck}$  such that  $(ck, x, (m_1, \dots, m_n)) \in R_L$  and the algorithm returns a proof  $\pi$ .
  - $Verify_{ck}(x, c_1, \dots, c_n, \pi)$  : It returns 0 (i.e. reject) or 1 (i.e. accept).



The protocol should satisfy three properties: correctness, knowledge soundness, and simulatability.



**Definition 3.1.2 (Correctness).** A commit-and-prove protocol proves statistical correctness with correctness error  $\rho : \mathbb{N} \rightarrow [0, 1]$  if for all adversaries  $\mathcal{A}$  :

$$\Pr \left[ ck \leftarrow Gen(1^\lambda); (x, m_1, r_1, \dots, m_n, r_n) \leftarrow \mathcal{A}(ck); c_i = Com_{ck}(m_i; r_i); \right. \\ \left. \pi \leftarrow Prove_{ck}(x, (m_1, r_1), \dots, (m_n, r_n)) : Verify_{ck}(x, c_1, \dots, c_n, \pi) = 0 \right] \leq \rho(\lambda)$$

where  $m_i \in \mathcal{M}_{ck}, r_i \in \mathcal{R}_{ck}$  so that  $(ck, x, (m_1, \dots, m_n)) \in R_L$ .

**Definition 3.1.3 (Knowledge Soundness).** A commit-and-prove protocol proves knowledge soundness with knowledge error  $\epsilon : \mathbb{N} \rightarrow [0, 1]$  if for all probabilistic polynomial-time algorithm  $\mathcal{A} \exists$  efficient extractor  $\mathcal{E}$  so that:

$$\Pr \left[ ck \leftarrow Gen(1^\lambda); (x, c_1, \dots, c_n, \pi) \leftarrow \mathcal{A}(ck); ((m_1^*, r_1^*), \dots, (m_n^*, r_n^*)) \leftarrow \mathcal{E}(c_1, \dots, c_n) : \right. \\ \left. Verify_{ck}(x, c_1, \dots, c_n, \pi) = 1 \wedge ((ck, x, (m_1^*, \dots, m_n^*)) \in R_L \vee \exists i, Com_{ck}(m_i^*, r_i^*) \neq c_i) \right] \leq \epsilon(\lambda)$$

where  $\mathcal{E}$  returns  $m_i^* \in \mathcal{M}_{ck}$  and  $r_i^* \in \mathcal{R}_{ck}$ .

**Definition 3.1.4 (Simulatability).** A commit-and-prove protocol is simulatable if there exist probabilistic polynomial-time simulators  $SimCom$  and  $SimProve$  such that for all probabilistic polynomial-time adversaries  $\mathcal{A}$  and commitment key  $ck \leftarrow$



$Gen(1^\lambda)$ :

$\Pr[(x, m_1, \dots, m_n) \leftarrow \mathcal{A}(ck); c_1, \dots, c_n \leftarrow SimCom_{ck}(x);$

$\pi \leftarrow SimProve_{ck}(x, c_1, \dots, c_n) : (ck, x, (m_1, \dots, m_n)) \in R_L \wedge \mathcal{A}(c_1, \dots, c_n, \pi) = 1]$

$\approx \Pr[(x, m_1, \dots, m_n) \leftarrow \mathcal{A}(ck); r_1 \dots r_n \leftarrow \xi; \forall i, c_i = Com_{ck}(m_i, r_i);$

$\pi \leftarrow Prove_{ck}(x, (m_1, r_1), \dots, (m_n, r_n)) : (ck, x, (m_1, \dots, m_n)) \in R_L \wedge \mathcal{A}(c_1, \dots, c_n, \pi) = 1]$

where  $\xi$  is a distribution on  $\mathcal{R}_{ck}$ .

### 3.2 Main Protocol

With knowledge of commit-and-prove protocol, we can focus on our main protocol defined in the figure below. This protocol uses the technique which is so-called ABDLOP commitment scheme. That is, Given secret information  $(\mathbf{s}_1, \mathbf{m})$  with  $\|\mathbf{s}_1\|$  small, we commit information as

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

with public random matrices  $\mathbf{A}_1, \mathbf{A}_2, \mathbf{B}$ . Define  $\sigma$ -trace map  $\text{Tr} : \mathcal{R}_q \mapsto \mathcal{R}_q$  as  $\text{Tr}(f) = (f + \sigma_{-1}(f))/2$  and we utilize lemma below.

**Lemma 3.2.1.** [LNP22, Lemma 4.8] Given  $\mathbf{s}_1 \in \mathcal{R}_q^{m_1}, \mathbf{m} \in \mathcal{R}_q^\ell$  and define  $\mathbf{s} = (\mathbf{s}_1, \sigma_{-1}(\mathbf{s}_1), \mathbf{m}, \sigma_{-1}(\mathbf{m}))$ , for any  $2(m_1 + \ell)$ -variate quadratic polynomial  $f : \mathcal{R}_q^{2(m_1 + \ell)} \rightarrow \mathcal{R}_q$  of the form  $f(\mathbf{x}) = \mathbf{x}^t \mathbf{R}_2 \mathbf{x} + \mathbf{r}_1^t \mathbf{x} + r_0$ , we set  $\text{Tr}(f)(\mathbf{x})$  to be  $\mathbf{x}^t \left( \frac{\mathbf{R}_2 + \mathbf{U}^t \sigma_{-1}(\mathbf{R}_2) \mathbf{U}}{2} \right) \mathbf{x} +$

$(\frac{\mathbf{r}_1^t \sigma_{-1}(\mathbf{r}_1^t) \mathbf{U}}{2}) \mathbf{x} + (r_0 + \sigma_{-1}(r_0))/2$ , where

$$\mathbf{U} = \begin{bmatrix} \mathbf{0} & \mathbf{I}_{km_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{I}_{km_1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_{k\ell} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{k\ell} & \mathbf{0} \end{bmatrix}.$$



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

Fix  $\sigma \in \text{Aut}(\mathcal{R})$  to be  $\sigma_{-1}$  and let  $\mathbf{x}_1 \in \mathcal{R}_q^{2m_1}$ ,  $\mathbf{x}_2 = (\mathbf{x}_{2,1}, \mathbf{x}_{2,2}) \in \mathcal{R}_q^{2(\lambda+\ell)}$ . Denote  $\mathbf{x}_{2,1} = (x_{2,1}^{(m)}, x_{1,1}^{(g)}, \dots, x_{1,\lambda/2}^{(g)})$  and  $\mathbf{x}_{2,2} = (x_{2,2}^{(m)}, x_{2,1}^{(g)}, \dots, x_{2,\lambda/2}^{(g)})$  and define  $\mathbf{x}_2^{(m)} = (x_{2,1}^{(m)}, x_{2,2}^{(m)})$ . Therefore, we can set  $f_j(\mathbf{x}_1, \mathbf{x}_2) := f_j(\mathbf{x}_1, \mathbf{x}_2^{(m)})$  for  $j \in [N]$ .

In order to check well-formedness of  $h_1, \dots, h_{\frac{\lambda}{2}}$ , we define

$$\begin{aligned} f_{N+k}(\mathbf{x}_1, \mathbf{x}_2) &= x_{1,k}^{(g)} + \text{Tr} \left( \sum_{u=1}^M \gamma_{2k-1,u} f_u \right) (\mathbf{x}_1, \mathbf{x}_2^{(m)}) \\ &\quad + X^{d/2} \text{Tr} \left( \sum_{u=1}^M \gamma_{2k,u} f_u \right) (\mathbf{x}_1, \mathbf{x}_2^{(m)}) - h_k \end{aligned} \quad (1)$$

and prove that  $f_{N+k}(\mathbf{s}_1, \sigma(\mathbf{s}_1), \mathbf{m}, \sigma(\mathbf{m})) = 0$  for  $1 \leq k \leq \frac{\lambda}{2}$ .

Private information:  $(\mathbf{s}_1, \mathbf{m}) \in \mathcal{R}_q^{m_1+\ell}$  so that  $\|\mathbf{s}_1\| \leq \alpha$ , randomness  $\mathbf{s}_2 \leftarrow \chi^{m_2}$

Public information:  $\mathbf{A}_1 \in \mathcal{R}_q^{n \times m_1}$ ,  $\mathbf{A}_2 \in \mathcal{R}_q^{n \times m_2}$ ,  $\mathbf{B} \in \mathcal{R}_q^{\ell \times m_2}$  such that

$$\begin{pmatrix} \mathbf{t}_A \\ \mathbf{t}_B \end{pmatrix} = \begin{pmatrix} \mathbf{A}_1 \\ \mathbf{0} \end{pmatrix} \mathbf{s}_1 + \begin{pmatrix} \mathbf{A}_2 \\ \mathbf{B} \end{pmatrix} \mathbf{s}_2 + \begin{pmatrix} \mathbf{0} \\ \mathbf{m} \end{pmatrix}, \mathbf{B}_g \in \mathcal{R}_q^{\lambda/2 \times m_2}, \mathbf{b} \in \mathcal{R}_q^{m_2}, \text{quadratic}$$

polynomials  $F_1, \dots, F_N$  and polynomial evaluations with vanishing constant

$$\text{coefficients } f_1, \dots, f_M : \mathcal{R}_q^{k(m_1+\ell)} \rightarrow \mathcal{R}_q, \sigma := \sigma_{-1} \in \text{Aut}(\mathcal{R})$$

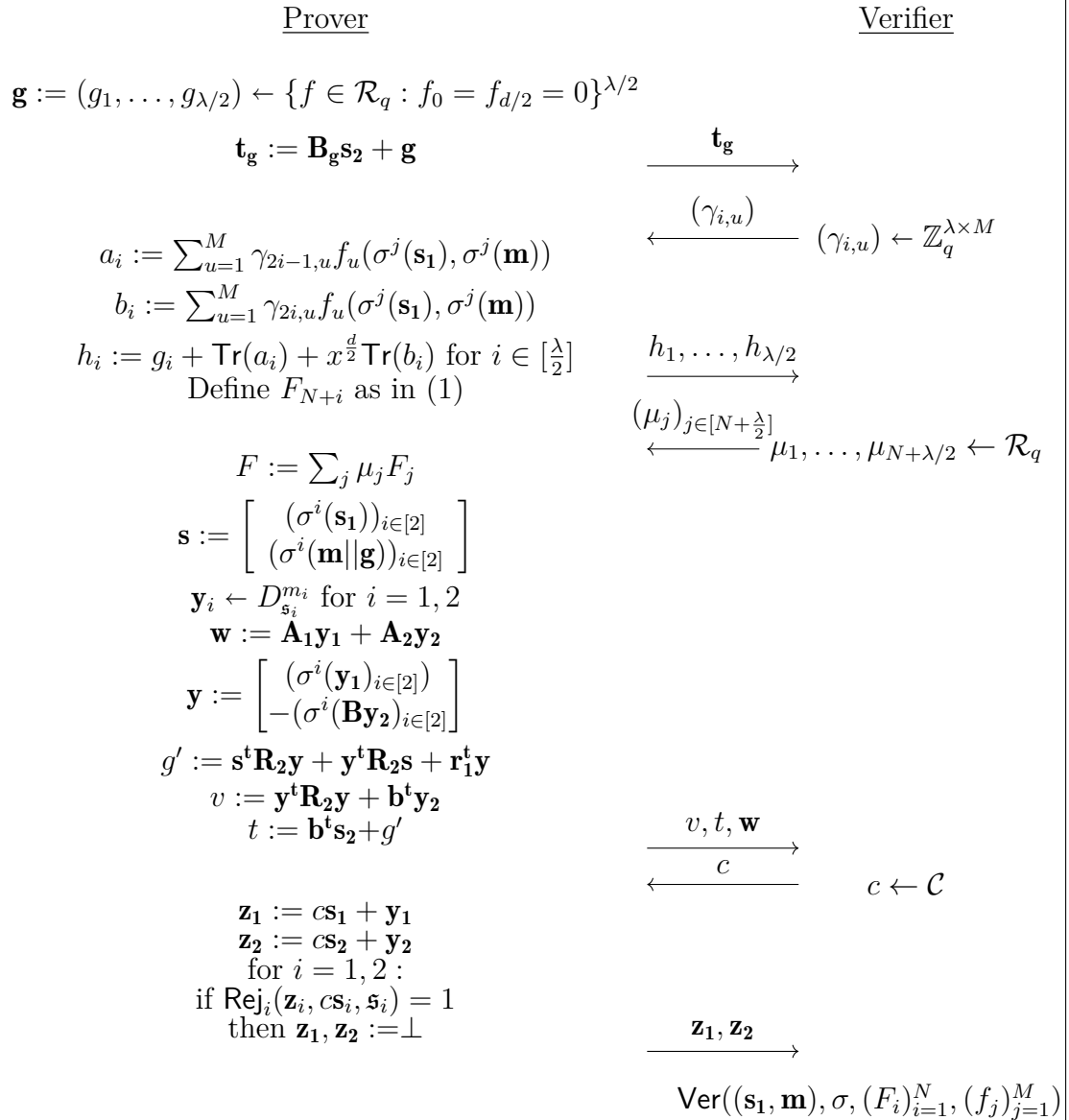


Figure 2: Commit-and-prove protocol  $\Pi((\mathbf{s}_2, \mathbf{s}_1, \mathbf{m}), \sigma, (F_i)_{i=1}^N, (f_j)_{j=1}^M)$  satisfying (i)  $\|\bar{c} \mathbf{s}_i\| \leq 2\bar{s}_i \sqrt{2m_i d}$  (ii)  $\mathbf{t}_A = \mathbf{A}_1 \mathbf{s}_1 + \mathbf{A}_2 \mathbf{s}_2$ ,  $\mathbf{t}_B = \mathbf{B} \mathbf{s}_2 + \mathbf{m}$  (iii)  $F_j(\sigma^i(\mathbf{s}_1), \sigma^i(\mathbf{m})) = 0$  for all  $j \leq N$  and (iv)  $f_j(\sigma^i(\mathbf{s}_1), \sigma^i(\mathbf{m})) = 0$  for all  $j \leq M$  where  $\mathbf{R}_2, \mathbf{r}_1, r_0$  is defined as  $F(\mathbf{x}) = \mathbf{x}^t \mathbf{R}_2 \mathbf{x} + \mathbf{r}_1^t \mathbf{x} + r_0$ .

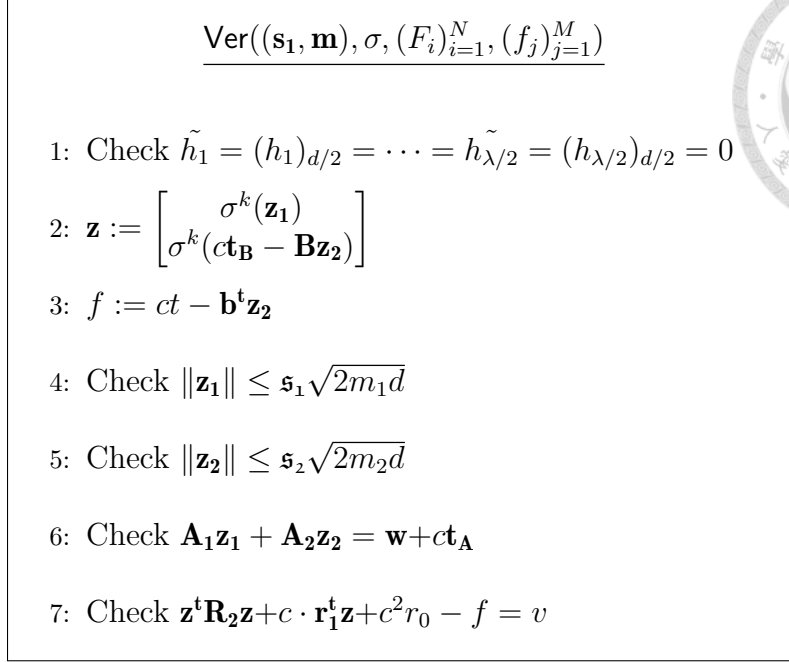


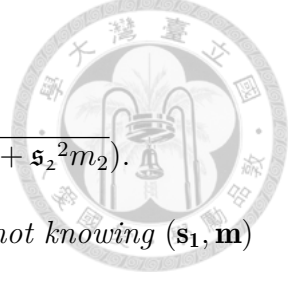
Figure 3: Verification equation for Figure 2

**Theorem 3.2.2** ([LNP22]). *If we select probability distribution  $\chi$  to be  $S_\nu$  and let  $\mathfrak{s}_1 = \gamma_1\alpha\eta$  as well as  $\mathfrak{s}_2 = \gamma_2\nu\eta\sqrt{m_2d}$  for some  $\gamma_1, \gamma_2 > 0$ . The parameter  $\eta$  is selected as described in Section 2.6. The protocol outlined in Figure 2 is commit-and-prove.*

*For correctness, if  $m_1, m_2 \geq \frac{640}{d}$ , then the probability that honest prover  $\mathcal{P}$  convinces the verifier  $\mathcal{V}$  is approximately  $1/(2 \exp(\frac{14}{\gamma_1} + \frac{1}{2\gamma_1^2} + \frac{1}{2\gamma_2^2}))$ .*

*For knowledge soundness, when given rewindable black-box access to a probabilistic prover  $\mathcal{P}^*$  convincing  $\mathcal{V}$ , there exists an efficient extractor  $\mathcal{E}$  either outputs  $(\mathbf{s}_2^*, \mathbf{s}_1^*, \mathbf{m}^*) \in \mathcal{R}_q^{m_1+m_2+\ell}$  and  $\bar{\mathbf{c}} \in \mathcal{R}_q^\times$  such that*

- $\begin{bmatrix} \mathbf{t}_A \\ \mathbf{t}_B \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{0} \end{bmatrix} \mathbf{s}_1^* + \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{B} \end{bmatrix} \mathbf{s}_2^* + \begin{bmatrix} \mathbf{0} \\ \mathbf{m}^* \end{bmatrix}$
- $F_j(\sigma^i(\mathbf{s}_1^*)_{i \in [2]}, \sigma^i(\mathbf{m}^*)_{i \in [2]}) = 0$  for  $j \in [N]$
- $f_j(\sigma^i(\mathbf{s}_1^*)_{i \in [2]}, \sigma^i(\mathbf{m}^*)_{i \in [2]}) = 0$  for  $j \in [M]$
- $\|\bar{\mathbf{c}}\|_\infty \leq 2\kappa$



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

or a  $\text{MSIS}_{n,m_1+m_2,B}$  solution for  $[\mathbf{A}_1 \mathbf{A}_2]$  with  $B = 8\eta\sqrt{2d(\mathfrak{s}_1^2m_1 + \mathfrak{s}_2^2m_2)}$ .

For commit-and-prove simulatability, there exists a simulator  $\mathcal{S}$  not knowing  $(\mathbf{s}_1, \mathbf{m})$  that returns a simulation of commitment  $(\mathbf{t}_A, \mathbf{t}_B)$  together with a non-aborting transcript between  $\mathcal{P}$  and  $\mathcal{V}$  so that for every adversary  $\mathcal{A}$  having advantage  $\epsilon$  in distinguishing  $\text{Sim}$  and  $(\mathbf{t}_A, \mathbf{t}_B)$  along with  $\text{View} \langle \mathcal{P}, \mathcal{V} \rangle$ , whenever the prover does not abort, there exists an algorithm  $\mathcal{A}'$  which distinguishes  $\text{Extended-MLWE}_{n+\ell+\lambda/2+1, m_2-n-\ell-\lambda/2-1, \chi, \mathcal{C}, \mathfrak{s}_2}$  with a probability of at least  $\epsilon/2 - (\frac{1}{2})^{128}$ .

*Proof.* We skip the proof of the knowledge soundness, since soundness follows identically as in [LNP22, Theorem 4.5].

**Correctness.** To begin with, we provide upper bounds for  $\|\mathbf{c}\mathbf{s}_1\|, \|\mathbf{c}\mathbf{s}_2\|$ . Based on Lemma 2.6.1 and the definition of our challenge space  $\mathcal{C}$ , we have  $\|\mathbf{c}\mathbf{s}_1\| \leq \eta\alpha$  and  $\|\mathbf{c}\mathbf{s}_2\| \leq \eta\nu\sqrt{m_2d}$ . The probability that both samplings  $\text{Rej}_1$  and  $\text{Rej}_2$  do not abort is at least

$$\frac{1 - 2^{-128}}{2 \cdot \exp(\frac{14}{\gamma_1} + \frac{1}{2\gamma_1^2}) \cdot \exp(\frac{1}{2\gamma_2^2})}$$

by Lemma 2.5.1. Additionally, we exploit Lemma 2.3.2 for  $t = \sqrt{2}$  and assumption  $m_1, m_2 \geq 640/d$ . Then,

$$\begin{aligned} \Pr\{\|\mathbf{z}_1\| > \mathfrak{s}_1\sqrt{2m_1d} \mid b = 0\} &\leq \Pr_{\mathbf{z} \leftarrow D_{\mathfrak{s}_1}^{m_1}}\{\|\mathbf{z}\| > \mathfrak{s}_1\sqrt{2m_1d}\} + 2^{-128} \text{ (by 2.5.1)} \\ &\leq \left(\frac{2}{e}\right)^{320} + 2^{-128} \text{ (by 2.3.2)} \end{aligned}$$

Similarly, the probability that  $\|\mathbf{z}_2\| \leq \mathfrak{s}_2\sqrt{2m_2d}$  is overwhelming. Since the honest prover knows genuine private  $\mathbf{s}_1, \mathbf{m}$ , the remaining verification equations must hold.

**Commit-and-prove Simulatability.** We show the statement by applying a hybrid

argument. We define simulator  $\mathcal{S}_0$  knowing private information  $\mathbf{s}_1, \mathbf{m}$ . For given challenges from the verifier, it yields honestly the commitment  $(\mathbf{t}_A, \mathbf{t}_B, \mathbf{t}_g, t)$  under randomness  $\mathbf{s}_2$  and generates  $h_1, \dots, h_{\lambda/2}$  honestly. Moreover, the algorithm samples masked opening  $\mathbf{z}_1 \leftarrow D_{\mathbf{s}_1}^{m_1}$  and  $\mathbf{z}_2 \leftarrow D_{\mathbf{s}_2}^{m_2}$  conditioned on  $\langle \mathbf{s}_2, \mathbf{z}_2 \rangle \geq 0$ . Ultimately, it assigns  $\mathbf{v} := \mathbf{z}^t \mathbf{R}_2 \mathbf{z} + c \mathbf{r}_1^t \mathbf{z} + c^2 r_0 - ct + \mathbf{b}^t \mathbf{z}_2$  and  $\mathbf{w} := \mathbf{A}_1 \mathbf{z}_1 + \mathbf{A}_2 \mathbf{z}_2 - c \mathbf{t}_A$ . By Lemma 2.5.1, the probability distribution of the simulation of the commitment together with the transcript generated by  $\mathcal{S}_0$  is statistically close to the real non-aborting one.

We set simulator  $\mathcal{S}_1$  still knowing the secret as follows. Simulator  $\mathcal{S}_1$  operates identically to  $\mathcal{S}_0$  but deviates by not honestly yielding the commitment  $(\mathbf{t}_A, \mathbf{t}_B, \mathbf{t}_g, t)$  and polynomials  $h_1, \dots, h_{\lambda/2}$ , it simulates polynomials by  $h_1, \dots, h_{\lambda/2} \leftarrow \{y \in \mathcal{R}_q : y_0 = y_{d/2} = 0\}$  uniformly and samples  $\mathbf{u} \leftarrow \mathcal{R}_q^{n+\ell+\lambda/2+1}$  and define

$$\begin{bmatrix} \mathbf{t}_A \\ \mathbf{t}_B \\ \mathbf{t}_g \\ t \end{bmatrix} = \mathbf{u} + \begin{bmatrix} \mathbf{A}_1 \mathbf{s}_1 \\ \mathbf{m} \\ \mathbf{g} \\ g' \end{bmatrix}. \quad (2)$$

**Lemma 3.2.3.** *If there exists a probabilistic polynomial-time adversary  $\mathcal{A}$  capable of distinguishing outputs from  $\mathcal{S}_0$  and  $\mathcal{S}_1$  with a probability of  $\epsilon$ , then there is a probabilistic polynomial-time adversary  $\mathcal{B}$  addressing the Extended-MLWE $_{n+\ell+\lambda/2+1, m_2-n-\ell-\lambda/2-1, \chi, \mathbf{C}, \mathbf{s}_2}$  with a probability of at least  $\epsilon/2$ .*

*Proof.* We define algorithm  $\mathcal{B}$  as follows. Given an Extended-MLWE tuple  $(\mathbf{C}, \mathbf{u}, \mathbf{z}_2, b)$ , where

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

$\mathcal{B}$  sets  $(\mathbf{t}_A, \mathbf{t}_B, \mathbf{t}_g, t)$  as in (2) and simulates the remaining components of the transcript the same as  $\mathcal{S}_0$  and  $\mathcal{S}_1$ . The algorithm yields the commitment and the tran-

script to  $\mathcal{A}$ . Assuming  $b = 1$ , when  $\mathbf{u} = \mathbf{C}\mathbf{s}_2$ , the output of  $\mathcal{B}$  follows the distribution of  $\mathcal{S}_0$ . Similarly, if  $u \leftarrow \mathcal{R}_q^{n+\ell+\lambda/2+1}$  uniformly, then the output comes from the distribution of  $\mathcal{S}_1$ . As a result, provided  $b = 1$ ,  $\mathcal{B}$  solves the Extended-MLWE problem with a probability of at least  $\epsilon$ . The statement holds since  $\Pr[b = 1] \geq \frac{1}{2}$ .  $\square$

On the other hand,  $h_1, \dots, h_{\lambda/2}$  are well-simulated because  $g_i \leftarrow \{x \in \mathcal{R}_q : x_0 = x_{d/2} = 0\}$  uniformly and  $\text{Tr}(a_i) + X^{d/2}\text{Tr}(b_i) \in \{x \in \mathcal{R}_q : x_0 = x_{d/2} = 0\}$  for every  $i \in [\lambda/2]$ .

Finally, the simulator  $\mathcal{S}$ , which lacks any knowledge of secret information, can be set to function identically to  $\mathcal{S}_1$ . Nevertheless, instead of generating commitment tuples  $(\mathbf{t}_A, \mathbf{t}_B, \mathbf{t}_g, t)$  as in (2), it samples  $(\mathbf{t}_A, \mathbf{t}_B, \mathbf{t}_g, t) \leftarrow \mathcal{R}_q^{n+\ell+\lambda/2+1}$ . Then, the distributions of  $\mathcal{S}$  and  $\mathcal{S}_1$  are totally the same.  $\square$

In the rest of this section, we focus on the size of the output from non-interactive proof obtained through the Fiat-Shamir transform of the protocol. Note that the messages  $v$  and  $\mathbf{w}$  for the non-interactive proof are not necessarily contained in the output because they can be uniquely determined by the remaining components. All challenges, except for  $c$ , can be computed as a hash of the preceding proof components. Thus, challenges require at most  $\lceil \log(2\kappa + 1) \rceil d$  bits. We have  $\mathbf{t}_A, \mathbf{t}_B, \mathbf{t}_g, t$  and  $h_i$ , full-sized elements in  $\mathcal{R}_q$ ; they require at most  $(n + \ell + \lambda + 1) \cdot \lceil \log q \rceil d$  bits. We encode the vectors  $\mathbf{z}_1, \mathbf{z}_2$  by exploiting Huffman coding. Concretely, suppose that  $z \leftarrow D_{\mathfrak{s}}$ , the discrete Gaussian distribution over  $\mathbb{Z}_q$ . We express  $z$  by  $z = z_1 \cdot 2^{\delta+1} + z_0$  where  $z_0 \equiv z \pmod{\pm 2^{\delta+1}}$ . The value of  $z_0$  is close to being uniformly distributed over  $\{-2^\delta, \dots, 2^\delta\}$ , for the expected absolute value of  $z$  is  $\mathfrak{s}$  and let  $2^\delta$  be near to  $\mathfrak{s}$ . The tails of the distribution of  $z_1$  decrease rapidly owing to tails of discrete Gaussian distribution. Therefore,  $z_0$  is distributed uniformly, which costs at most  $\delta + 1$  bits,



and we exploit Huffman coding to encode  $z_1$ . Next, suppose that  $\mathfrak{s} = 2^\delta$  and that the tails of  $z_1$  are identical to those in Gaussian distribution centered at 0. In this scenario, the previously described compression requires approximate 1.57 bits to represent  $z_1$  averagely. Hence, representation of  $z$  gives approximately  $2.57 + \log \mathfrak{s}$  bits. We apply this technique to  $\mathbf{z}_1, \mathbf{z}_2$ , and the whole commitment and proof length is about

$$(n+\ell+\lambda+1)d\lceil\log q\rceil+\lceil\log(2\kappa+1)\rceil d+m_1d\cdot(2.57+\lceil\log \mathfrak{s}_1\rceil)+m_2d\cdot(2.57+\lceil\log \mathfrak{s}_2\rceil) \text{ bits.}$$

## 4 Applications to Integer Relation



In this section, we present two efficient zero-knowledge proofs for integer addition and multiplication for committed information. The technique applied in this section is an adaptation of one in [LNS20]. In particular, we will elaborate in the subsection below that given commitments to integers  $a_i, c$ , we could prove  $\sum_{i=1}^k a_i = c$ . We will explain in the other subsection how to prove an integer multiplication  $a \cdot b = c$  under some restrictions for integers  $a, b, c$ .

### 4.1 Integer Addition

First, we consider integers  $a, b, c \in [-2^{N-1}, 2^{N-1} - 1]$  and we attempt to demonstrate  $a + b = c$ . We represent  $a$  by a binary vector  $\vec{a} = (a_0, \dots, a_{N-1})$  satisfying  $a = -a_{N-1}2^{N-1} + \sum_{i=0}^{N-2} a_i 2^i$  with  $a_0, \dots, a_{N-1} \in \{0, 1\}$ . We give identical representations for  $b$  together with  $c$  via adopting the same approach and obtain  $\vec{b}$  along with  $\vec{c}$ . We call this technique two's complement representation and denote  $\text{TC}(a) := \vec{a}$ ,  $\text{TC}(b) := \vec{b}$ ,  $\text{TC}(c) := \vec{c}$ . Then integer addition  $a + b = c$  is equivalent to  $a(x) + b(x) + f(x)(x - 2) = c(x)$  for a polynomial  $f(x) \in \mathbb{Z}[x]$ , according to Gauss' lemma.

By coefficient comparison, we have the following system of linear equations.



$$\left\{ \begin{array}{l} a_0 + b_0 = 2f_0 + c_0 \\ a_1 + b_1 + f_0 = 2f_1 + c_1 \\ \vdots \\ a_{N-2} + b_{N-2} + f_{N-3} = 2f_{N-2} + c_{N-2} \\ a_{N-1} + b_{N-1} - f_{N-2} = -2f_{N-1} + c_{N-1} \\ f_{N-1} = 0 \end{array} \right.$$

**Lemma 4.1.1.** *Let  $f(x)$  be defined as above. Then for each coefficient  $f_j$  of  $f(x)$  corresponding to  $x^j$ ,  $f_j \in \{0, 1\}$ .*

*Proof.* For  $f_{N-1}$  case, it is apparent from the system of linear equation. Consider  $f_0$ ,  $-1 \leq 2f_0 = a_0 + b_0 - c_0 \leq 2$  and  $f_0 \in \mathbb{Z}$  imply  $f_0 \in \{0, 1\}$ . For  $i = 1, \dots, N-2$ ,  $2f_i = a_i + b_i - c_i + f_{i-1}$  and, by induction hypothesis,

$$-1 \leq -1 + f_{i-1} \leq a_i + b_i - c_i + f_{i-1} \leq 2 + f_{i-1} \leq 3,$$

and thus  $f_i \in \{0, 1\}$  for all  $i$ . □

All the aforementioned coefficients are small, we can consider those coefficients modulo  $q$ , leading to an equivalent expression in vector notation:

$$\vec{a} + \vec{b} + E\vec{f} \equiv 2J\vec{f} + \vec{c} \pmod{q} \quad (*)$$

with coefficient vector  $\vec{f}$  of  $f(x)$ ,

$$E = \begin{pmatrix} 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ & & & \vdots & \\ 0 & 0 & \dots & -1 & 0 \end{pmatrix}, \text{ and } J = \begin{pmatrix} 1 & & & & 0 \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \\ 0 & & & & -1 \end{pmatrix} \in \mathbb{Z}_q^{N \times N}$$



Observe that  $E - 2J$  is a lower triangular matrix and  $\det(E - 2J) = -2^N$ , which implies  $E - 2J$  is invertible. Recall that if  $\vec{r} = (r_0, \dots, r_{d-1}), \vec{s} = (s_0, \dots, s_{d-1}) \in \mathbb{Z}_q^d$  and corresponding polynomials  $r(X) = \sum_{i=0}^{d-1} r_i X^i, s(X) = \sum_{i=0}^{d-1} s_i X^i$ , then

$$\begin{aligned} \langle \vec{r}, \vec{s} \rangle \pmod{q} &= \sum_{n=0}^{d-1} r_n \cdot s_n = r_0 s_0 + \sum_{n=1}^{d-1} (-1)(-r_n) s_n \\ &= r_0 s_0 + \sum_{n=1}^{d-1} (-r_n) s_n X^d \\ &= r_0 s_0 + \sum_{n=1}^{d-1} (-r_n) X^{d-n} s_n X^n \\ &= r_0 s_0 + \sum_{n=1}^{d-1} r_n (X^{-1})^n \cdot s_n X^n \\ &= r(X^{-1}) \cdot s(X)|_{X=0} = \sigma_{-1}(r(X)) \cdot s(X)|_{X=0} \end{aligned}$$

The fact that  $X^d \equiv -1 \in \mathcal{R}_q$  is followed by the third equality and the fifth equality follows  $(X^{-1})^n = (-X^{d-1})^n = (-1)^n X^{(n-1)d+(d-n)} = (-1)^n (-1)^{n-1} X^{d-n} = -X^{d-n}$ .

In general,  $\vec{r}, \vec{s} \in \mathbb{Z}_q^{nd}$ ; we can define corresponding polynomial vectors naturally  $\mathbf{r} = (r_1, \dots, r_n), \mathbf{s} = (s_1, \dots, s_n) \in \mathcal{R}_q^n$  with coefficients as  $\vec{r} = (r_{1,0}, \dots, r_{n,d-1}), \vec{s} =$

$(s_{1,0}, \dots, s_{n,d-1})$  in the same approach. That is, Thus, we have

$$\begin{aligned} \langle \vec{r}, \vec{s} \rangle \pmod{q} &= \sum_{i,j=1,0}^{n,d-1} r_{i,j} \cdot s_{i,j} = \sum_{i=1}^n \sum_{j=0}^{d-1} r_{i,j} \cdot s_{i,j} \\ &= \sum_{i=1}^n r_i(X^{-1}) \cdot s_i(X)|_{X=0} \\ &= \sum_{i=1}^n \sigma_{-1}(r_i(X)) \cdot s_i(X)|_{X=0} \end{aligned}$$



Remark that if  $\vec{s}$  is a public vector and  $\vec{r}$  is a secret vector, then  $\langle \vec{r}, \vec{s} \rangle$  is the constant coefficient of a linear function evaluating in  $\mathcal{R}_q$ , by the equation above. On the other hand, if both  $\vec{r}, \vec{s}$  are secret information, then the inner product is identical to the constant coefficient of a multivariate quadratic function.

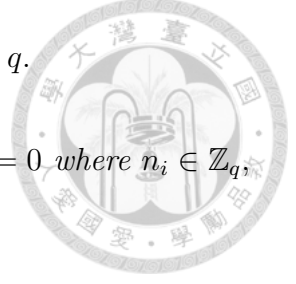
Now, we examine those secret vectors  $(\vec{a}, \vec{b}, \vec{c}, \vec{f})$ . There exists natural norm bound such that  $\|\vec{a}\|, \|\vec{b}\|$ , and  $\|\vec{c}\| \leq \sqrt{N}$ . Hence,  $\mathbf{s}_1 = (\vec{a}, \vec{b}, \vec{c})$  and  $\mathbf{m} = \vec{f}$  in our protocol. Equation (\*) is equivalent to  $\mathbf{R}_1 \mathbf{s}_1 + \mathbf{R}_m \mathbf{m} = \mathbf{0} \pmod{q}$  where  $\mathbf{R}_1 = [-I_N \parallel -I_N \parallel I_N]$  and  $\mathbf{R}_m = [E - 2J]$ . Furthermore,  $\vec{a}, \vec{b}, \vec{c}$ , and  $\vec{f}$  consist of binary elements, with the condition that  $f_{N-1} = 0$ , and can be defined by some multivariate quadratic polynomials and linear functions. The defining multivariate quadratic polynomials with vanishing constant coefficients to prove those vectors are binary vectors are as follows:

$$G_a(\sigma^i(\mathbf{s}_1), \sigma^i(\mathbf{m})) = \langle \vec{a}, \vec{a} - \vec{1} \rangle \pmod{q}$$

$$G_b(\sigma^i(\mathbf{s}_1), \sigma^i(\mathbf{m})) = \langle \vec{b}, \vec{b} - \vec{1} \rangle \pmod{q}$$

$$G_c(\sigma^i(\mathbf{s}_1), \sigma^i(\mathbf{m})) = \langle \vec{c}, \vec{c} - \vec{1} \rangle \pmod{q}$$

$$G_f(\sigma^i(\mathbf{s}_1), \sigma^i(\mathbf{m})) = \langle \vec{f}, \vec{f} - \vec{\chi} \rangle \pmod{q}$$



where  $\vec{\chi} = (1, \dots, 1, 0)$  and provided that  $N$  satisfies  $2(N - 1) < q$ .

**Lemma 4.1.2.** *If  $j < d - 1$  and  $s \in \mathcal{R}_q$  satisfying  $\prod_{i=n_0}^{n_j} (s - i) = 0$ , where  $n_i \in \mathbb{Z}_q$ , then  $s \in \mathbb{Z}_q x^{\frac{d}{2}} + \mathbb{Z}_q \subset \mathcal{R}_q$ .*

*Proof.* Recall that  $\mathbb{Z}_q[x] \xrightarrow{\sim} \bigoplus_{i=1}^n \mathbb{Z}_{q_i}[x]$  by Chinese Remainder Theorem. We denote  $\mathcal{R}_{q_i}$  to be  $\mathbb{Z}_{q_i}[x]/(x^d+1)$  for  $i \leq n$ . We have  $\mathcal{R}_q \xrightarrow{\sim} \bigoplus_{i=1}^n \mathcal{R}_{q_i} \xrightarrow{\sim} \bigoplus_{i=1}^n \mathbb{Z}_{q_i}[x]/(x^{\frac{d}{2}} - \zeta_i) \oplus \mathbb{Z}_{q_i}[x]/(x^{\frac{d}{2}} + \zeta_i)$ . According to the canonical isomorphism,  $0 = \prod_{i=0}^j (s - n_i) \in \mathcal{R}_q \mapsto ((0, 0), \dots, (0, 0)) \in \bigoplus_{i=1}^n \mathbb{Z}_{q_i}[x]/(x^{\frac{d}{2}} - \zeta_i) \oplus \mathbb{Z}_{q_i}[x]/(x^{\frac{d}{2}} + \zeta_i)$ .

$\mathbb{Z}_{q_i}[x]/(x^{\frac{d}{2}} \pm \zeta_i)$  are fields, so  $s$  is an constant in  $\mathbb{Z}_{q_i}[x]/(x^{\frac{d}{2}} - \zeta_i)$ ,  $\mathbb{Z}_{q_i}[x]/(x^{\frac{d}{2}} + \zeta_i)$  respectively. Again, we recover  $s$  by  $\mathcal{R}_{q_i} \xleftarrow{\sim} \mathbb{Z}_{q_i}[x]/(x^{\frac{d}{2}} - \zeta_i) \oplus \mathbb{Z}_{q_i}[x]/(x^{\frac{d}{2}} + \zeta_i)$  via  $s \pmod{q_i} \leftarrow (\alpha_i, \beta_i)$ . That is, we need to solve  $s = \sum_{j=0}^{d-1} s_j x^j \in \mathcal{R}_{q_i}$  such that  $s \equiv \sum_{j=0}^{\frac{d}{2}-1} (s_j + s_{\frac{d}{2}+j} \zeta_i) x^j \equiv \alpha_i \pmod{x^{\frac{d}{2}} - \zeta_i}$  (resp.  $\sum_{j=0}^{\frac{d}{2}-1} (s_j - s_{\frac{d}{2}+j} \zeta_i) x^j \equiv \beta_i \pmod{x^{\frac{d}{2}} + \zeta_i}$ ). By comparing coefficient, we obtain  $\forall j \neq 0, s_j = s_{\frac{d}{2}+j} = 0, s_0 + s_{\frac{d}{2}} \zeta_i = \alpha_i$ , and  $s_0 - s_{\frac{d}{2}} \zeta_i = \beta_i$ . Thus  $s \in \mathbb{Z}_{q_i} x^{\frac{d}{2}} + \mathbb{Z}_{q_i} \subset \mathcal{R}_{q_i}$ . Finally, according to canonical  $\mathbb{Z}_q[x] \xleftarrow{\sim} \bigoplus_{i=1}^n \mathbb{Z}_{q_i}[x]$ , we have  $s \in \mathcal{R}_q$  is of the desired form.  $\square$

Applying the lemma, we can set defining quadratic polynomials and quadratic polynomials having zero constant coefficients for secret  $(\mathbf{s}_1, \mathbf{m})$  as given below:

$$F_{a_i}(\sigma^j(\mathbf{s}_1), \sigma^j(\mathbf{m})) = a_i(a_i - 1), G_{a_i}(\sigma^j(\mathbf{s}_1), \sigma^j(\mathbf{m})) = \langle x^{\frac{d}{2}}, a_i \rangle$$

$$F_{b_i}(\sigma^j(\mathbf{s}_1), \sigma^j(\mathbf{m})) = b_i(b_i - 1), G_{b_i}(\sigma^j(\mathbf{s}_1), \sigma^j(\mathbf{m})) = \langle x^{\frac{d}{2}}, b_i \rangle$$

$$F_{c_i}(\sigma^j(\mathbf{s}_1), \sigma^j(\mathbf{m})) = c_i(c_i - 1), G_{c_i}(\sigma^j(\mathbf{s}_1), \sigma^j(\mathbf{m})) = \langle x^{\frac{d}{2}}, c_i \rangle$$

$$F_{f_i}(\sigma^j(\mathbf{s}_1), \sigma^j(\mathbf{m})) = f_i(f_i - 1), G_{f_i}(\sigma^j(\mathbf{s}_1), \sigma^j(\mathbf{m})) = \langle x^{\frac{d}{2}}, f_i \rangle$$

for all  $i \in \{0 \dots N - 1\}$ .

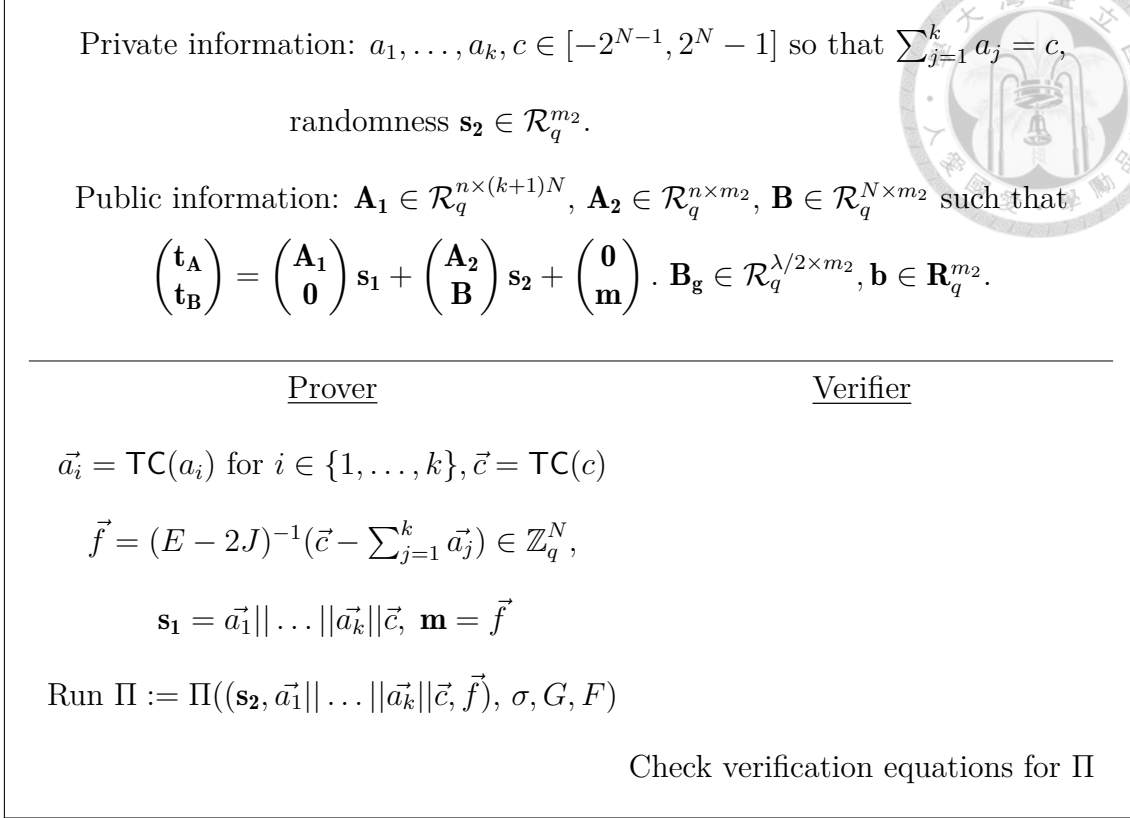


Figure 4: Proof of general integer addition where  $2k < d$ .

Generally speaking, if one attempts to show an addition of several integers. That is, one want to show that  $\sum_{i=1}^k a_i = c$  for some  $i \in \mathbb{N}$  and  $a_i, c \in [-2^{N-1}, 2^N - 1]$ . The origin addition problem is equivalent to  $\sum_{i=1}^k a_i(x) + f(x)(x - 2) = c(x)$  for some integer polynomial  $f(x)$  as we mentioned before. Let  $a_i(x) = -a_{i,N-1}x^{N-1} +$

$\sum_{j=0}^{N-2} a_{i,j}x^j$  for  $i \in [k]$ . Therefore, we obtain the system of linear equations below.

$$\left\{ \begin{array}{l} \sum_{i=1}^k a_{i,0} = 2f_0 + c_0 \\ \sum_{i=1}^k a_{i,1} + f_0 = 2f_1 + c_1 \\ \vdots \\ \sum_{i=1}^k a_{i,N-2} + f_{N-3} = 2f_{N-2} + c_{N-2} \\ \sum_{i=1}^k a_{i,N-1} - f_{N-2} = -2f_{N-1} + c_{N-1} \\ f_{N-1} = 0 \end{array} \right.$$



We prove the following equality for this protocol:

$$\sum_{i=1}^k \vec{a}_i + E\vec{f} \equiv 2J\vec{f} + \vec{c} \pmod{q} \quad (**)$$

with matrices  $E$  and  $J$  as usual. Here, we need to apply a lemma to the system of linear equations to ensure that the equality for polynomials still holds when  $(**)$  holds, provided that  $q$  is large enough.

**Lemma 4.1.3.** *If binary vectors  $\vec{a}_i = (a_{i,j})$ ,  $\vec{c} = (c_j)$ , and integer vector  $\vec{f} = (f_j)$  for  $j \in \{0, \dots, N-1\}$  satisfying the above linear equations, then  $0 \leq f_j \leq k \cdot (2 - \frac{1}{2^j})$  for all  $j$ .*

*Proof.* As  $j = 0$ ,  $-1 \leq 2f_0 = \sum_{i=1}^k a_{i,0} - c \leq k$ . It implies  $-\frac{1}{2} < 0 \leq f_0 \leq \frac{k}{2} < k \cdot (2 - 1) = k$ , since  $f_0 \in \mathbb{Z}$ . For  $j = 1, \dots, N-3$ , we assume that  $0 \leq f_j \leq (2 - \frac{1}{2^j})k$ .



Since  $-1 \leq \sum_{i=1}^k a_{i,j+1} - c_{j+1} = 2f_{j+1} - f_j \leq k$ ,  $-1/2 \leq f_{j+1} - f_j/2 \leq k/2$ .

$$\begin{aligned} -\frac{1}{2} + f_j/2 &\leq f_{j+1} = f_j/2 + f_{j+1} - f_j/2 \leq k/2 + f_j/2 \\ -\frac{1}{2} &\leq f_{j+1} \leq (2 - \frac{1}{2^j})k/2 + k/2 \text{ (by induction hypothesis)} \\ 0 &\leq f_{j+1} \leq (1 - \frac{1}{2^{j+1}})k + k/2 = (3/2 - \frac{1}{2^{j+1}})k. \end{aligned}$$



For  $j = N - 1$ , we have  $f_{N-1} = 0$ . Hence,  $0 \leq f_j \leq k \cdot (2 - \frac{1}{2^j})$  by induction.  $\square$

It is evident that  $\|\vec{a}_i\| \leq \sqrt{N}$  and  $\|c\| \leq \sqrt{N}$  with  $\vec{a}_i = \text{TC}(a_i)$  and  $\vec{c} = \text{TC}(c)$ . We can utilize the above results to define our secret information  $(\mathbf{s}_1, \mathbf{m})$  with  $\mathbf{s}_1 := (\vec{a}_1, \dots, \vec{a}_k, \vec{c})$  along with  $\mathbf{m} := \vec{f}$ , provided  $2k < d$ . It is straightforward that  $\|\mathbf{s}_1\| \leq \sqrt{(k+1)N}$ , assuming that  $\mathbf{MSIS}_{n,(k+1)N, \sqrt{(k+1)N}}$  is hard. With the goal of proving integer addition  $\sum_{i=1}^k a_i = c$ , we define multivariate quadratic polynomials  $G_{\vec{a}_i}(\sigma^j(\mathbf{s}_1), \sigma^j(\mathbf{m})) = \langle \vec{a}_i, \vec{a}_i - \vec{1} \rangle$ , as well as  $G_{\vec{c}}$  for  $i \in [k]$ . Moreover, needing to show  $f_{N-1} = 0$ , we define  $F_f(\sigma^j(\mathbf{s}_1), \sigma^j(\mathbf{m})) = f_{N-1}$ . Similarly, we rewrite equation (\*\*) to be  $[\mathbf{R}_1 || \mathbf{R}_m] \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{m} \end{bmatrix} = \mathbf{0}$  with  $\mathbf{R}_1 = [-I_N || \dots || -I_N || I_N]$  together with  $\mathbf{R}_m$  as usual.

On the other hand, we have quadratic polynomials to show  $(\mathbf{s}_1, \mathbf{m})$  are integer vectors  $F_{a_{i,j}}(\sigma^k(\mathbf{s}_1), \sigma^k(\mathbf{m})) = a_{i,j}(a_{i,j} - 1)$ ,  $G_{a_{i,j}}(\sigma^k(\mathbf{s}_1), \sigma^k(\mathbf{m})) = \langle x^{\frac{d}{2}}, a_{i,j} \rangle$  along with  $F_{c_j}, G_{c_j}$  for  $(i, j) \in [k] \times \{0, \dots, N-1\}$ . For  $\vec{f}$ , we set defining polynomials to be  $G_{f_{i,j}}(\sigma^k(\mathbf{s}_1), \sigma^k(\mathbf{m})) = \langle x^j, f_i \rangle$  for  $i \in \{0, \dots, N-1\}, 0 < j < d$ . Now, we define the set of polynomials  $F := \{F_{a_{i,\ell}}, F_{c_\ell}, F_f : \ell \in \{0, \dots, N-1\}, i \in [k]\}$  and the set  $G := \{G_1, \dots, G_N, G_{a_{i,j}}, G_{c_j}, G_{f_j,\ell}, G_{\vec{a}_i}, G_{\vec{c}} : j \in \{0, \dots, N-1\}, i \in [k], 0 < \ell < d\}$  of polynomials of degree at most 2 such that evaluations of those polynomials have vanishing constant coefficients where  $G_1 \dots, G_N$  are linear functions corresponding



to the row of  $[\mathbf{R}_1 || \mathbf{R}_m]$ .

**Proof size.** As we discussed previously, the proof size has nothing to do with the number of equations that we want to prove, whence the proof size for general integer addition is about

$$(n+N+\lambda+1)d\lceil\log q\rceil + \lceil\log(2\kappa+1)\rceil d + (k+1)Nd \cdot (2.57 + \lceil\log \mathfrak{s}_1\rceil) + m_2 d \cdot (2.57 + \lceil\log \mathfrak{s}_2\rceil)$$

bits where  $\mathfrak{s}_1 = \gamma_1 \eta \sqrt{(k+1)N}$  and  $\mathfrak{s}_2 = \gamma_2 \eta \nu \sqrt{m_2 d}$ .

## 4.2 Integer Multiplication

Given integers  $c \in [-2^{2N-1}, 2^{2N} - 1]$  and  $a, b \in [-2^{N-1}, 2^N - 1]$  satisfying  $ab = c$ , we apply the polynomial approach to prove the desired integer multiplication. Precisely speaking, it suffices to show that  $a(x) \cdot b(x) + f(x)(x-2) = c(x)$  and that the coefficients of  $a(x), b(x), c(x)$  are either 1 or 0. By the proof of Lemma 4.1.3, we have  $0 \leq \|f\|_\infty \leq N + 1$ . Without loss of generality, we suppose that  $N = d/2$ . Then, we set private  $\mathbf{s}_1$  to be  $(a(x), a_0, \dots, a_{d/2-1}, b(x), b_0, \dots, b_{d/2-1}, c(x), c_0, \dots, c_{d-1})$  with  $a_i, b_i, c_i$  are coefficients of  $a(x), b(x), c(x)$  respectively. Similarly, define secret  $\mathbf{m}$  to be  $(f(x), f_0, \dots, f_{d-1})$  where  $f_{d-1} = 0$ . It is evident that  $\|\mathbf{s}_1\| \leq \sqrt{d + d + 2d} = 2\sqrt{d}$ .

Now we set the defining polynomials for our private information  $(\mathbf{s}_1, \mathbf{m})$ . To ensure well-formedness of  $a(x), b(x), c(x)$ , we define  $F_0(\sigma^i(\mathbf{s}_1), \sigma^i(\mathbf{m})) = a(x) \cdot b(x) - c(x) + f(x) \cdot (x-2)$  along with  $F_a(\sigma^i(\mathbf{s}_1), \sigma^i(\mathbf{m})) = \sum_{j=0}^{d/2-2} a_j x^j - a_{d/2-1} x^{d/2-1} - a(x)$  and similarly for  $F_b, F_c, F_f$ . Next, we are necessary to show that  $a_i, b_i, c_j, f_j$  for  $0 \leq i \leq d/2 - 1, 0 \leq j \leq d - 1$  are constants. Besides, we still need to prove that  $a_i, b_i, c_j$  are either 1 or 0 for all  $i, j$  and that  $f_{d-1} = 0$ . Hence, we set  $F_{a_i}(\sigma^k(\mathbf{s}_1), \sigma^k(\mathbf{m})) = a_i(a_i - 1), G_{a_i}(\sigma^k(\mathbf{s}_1), \sigma^k(\mathbf{m})) = \langle a_i, x^{\frac{d}{2}} \rangle$  together with  $G_a(\sigma^i(\mathbf{s}_1), \sigma^i(\mathbf{m})) = \langle \text{TC}(a), \text{TC}(a) -$

1) and similarly for  $F_{c_1}, G_{c_1}, \dots, F_{c_{d-1}}, G_{c_{d-1}}, G_c, F_{b_i}, G_{b_i}, G_b$  where  $0 \leq i \leq d/2 - 1$ . Furthermore, we consider polynomials for  $(f_0, \dots, f_{d-1})$ . That is, for  $0 \leq i \leq d-2$ ,  $G_{f_i, j}(\sigma^k(\mathbf{s}_1), \sigma^k(\mathbf{m})) = \langle f_i, x^j \rangle$  together with  $F_{f_{d-1}} = f_{d-1}$ . Next, we define sets  $F := \{F_0, F_a, F_b, F_c, F_f, F_{a_i}, F_{b_i}, F_{c_j}\}$  along with  $G := \{G_a, G_b, G_c, G_{a_i}, G_{b_i}, G_{c_j}, G_{f_\ell, k} : 0 \leq \ell < d-1, 0 < k < d\}$  respectively; then, we use the protocol to prove that private  $(\mathbf{s}_1, \mathbf{m})$  satisfying  $h(\sigma^k(\mathbf{s}_1), \sigma^k(\mathbf{m})) = 0 = \tilde{g}(\sigma^k(\mathbf{s}_1), \sigma^k(\mathbf{m}))$  for all  $h \in F, g \in G$ .

<p>Private information: <math>a, b \in [-2^{N-1}, 2^N - 1], c \in [-2^{2N-1}, 2^{2N} - 1]</math> so that</p> $a \cdot b = c, \text{ randomness } \mathbf{s}_2 \in \mathcal{R}_q^{m_2}.$ <p>Public information: <math>\mathbf{A}_1 \in \mathcal{R}_q^{n \times (k+1)N}, \mathbf{A}_2 \in \mathcal{R}_q^{n \times m_2}, \mathbf{B} \in \mathcal{R}_q^{N \times m_2}</math> such that</p> $\begin{pmatrix} \mathbf{t}_A \\ \mathbf{t}_B \end{pmatrix} = \begin{pmatrix} \mathbf{A}_1 \\ \mathbf{0} \end{pmatrix} \mathbf{s}_1 + \begin{pmatrix} \mathbf{A}_2 \\ \mathbf{B} \end{pmatrix} \mathbf{s}_2 + \begin{pmatrix} \mathbf{0} \\ \mathbf{m} \end{pmatrix}. \mathbf{B}_g \in \mathcal{R}_q^{\lambda/2 \times m_2}, \mathbf{b} \in \mathbf{R}_q^{m_2}.$	
<u>Prover</u>	<u>Verifier</u>
$a(x) = -\text{TC}(a)_{N-1}x^{N-1} + \sum_{i=0}^{N-2} \text{TC}(a)_i x^i$ $b(x) = -\text{TC}(b)_{N-1}x^{N-1} + \sum_{i=0}^{N-2} \text{TC}(b)_i x^i$ $c(x) = -\text{TC}(c)_{2N-1}x^{2N-1} + \sum_{i=0}^{2N-2} \text{TC}(c)_i x^i$ $f(x) = (c(x) - a(x) \cdot b(x))/(x - 2)$ $\mathbf{s}_1 = (a(x) \parallel \text{TC}(a) \parallel b(x) \parallel \text{TC}(b) \parallel c(x) \parallel \text{TC}(c))$ $\mathbf{m} = (f(x) \parallel f_0 \parallel \dots \parallel f_{2N-1})$ $\text{Run } \Pi := \Pi((\mathbf{s}_2, \mathbf{s}_1, \mathbf{m}), \sigma, G, F)$	
Check verification equations for $\Pi$	

Figure 5: Proof of integer multiplication where  $N \leq d/2$ .

**Proof size.** As for the proof size, note that  $\|\mathbf{s}_1\| \leq 2\sqrt{2N}$ . Therefore, the total

proof length should be about

$$\lceil \log(2\kappa + 1) \rceil d + (n + 2N + \lambda + 2)d \lceil \log q \rceil + (4N + 3)d \cdot (2.57 + \lceil \log \mathfrak{s}_1 \rceil) + m_2 d \cdot (2.57 + \lceil \log \mathfrak{s}_2 \rceil) \text{ bits}$$

where  $\mathfrak{s}_1 = 2\gamma_1\eta\sqrt{2N}$  as well as  $\mathfrak{s}_2 = \gamma_2\eta\nu\sqrt{m_2d}$ .

To recap, we improved the method of proving that an element in  $\mathcal{R}_q$  is indeed a constant in  $\mathbb{Z}_q$  by indicating that  $\mathcal{R}_q$  behaves almost like a field in some situations. Next, through the application of Lemma 4.1.2, we have designed two efficient lattice-based zero-knowledge protocols based on result from [LNP22], [LNS20], specifically tailored for integer relations. One protocol is designed for solving arbitrary integer addition, while the other is tailored for addressing product of two integers.


## References

- [Ajt96] Miklós Ajtai. Generating hard instances of lattice problems. *Electron. Colloquium Comput. Complex.*, TR96, 1996.
- [ALS20] Thomas Attema, Vadim Lyubashevsky, and Gregor Seiler. Practical product proofs for lattice commitments. In Daniele Micciancio and Thomas Ristenpart, editors, *Advances in Cryptology – CRYPTO 2020*, pages 470–499, Cham, 2020. Springer International Publishing.
- [Ban93] W. Banaszczyk. New bounds in some transference theorems in the geometry of numbers. *Mathematische Annalen*, 296(1):625–635, Dec 1993.
- [BDL<sup>+</sup>18] Carsten Baum, Ivan Damgård, Vadim Lyubashevsky, Sabine Oechsner, and Chris Peikert. More efficient commitments from structured lattice



assumptions. In Dario Catalano and Roberto De Prisco, editors, *Security and Cryptography for Networks*, pages 368–385, Cham, 2018. Springer International Publishing.

- [BLS19] Jonathan Bootle, Vadim Lyubashevsky, and Gregor Seiler. Algebraic techniques for short(er) exact lattice-based zero-knowledge proofs. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology – CRYPTO 2019*, pages 176–202, Cham, 2019. Springer International Publishing.
- [ENS20] Muhammed F. Esgin, Ngoc Khanh Nguyen, and Gregor Seiler. Practical exact proofs from lattices: New techniques to exploit fully-splitting rings. In Shiho Moriai and Huaxiong Wang, editors, *Advances in Cryptology – ASIACRYPT 2020*, pages 259–288, Cham, 2020. Springer International Publishing.
- [ESLL19] Muhammed F. Esgin, Ron Steinfeld, Joseph K. Liu, and Dongxi Liu. Lattice-based zero-knowledge proofs: New techniques for shorter and faster constructions and applications. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology – CRYPTO 2019*, pages 115–146, Cham, 2019. Springer International Publishing.
- [LNP22] Vadim Lyubashevsky, Ngoc Khanh Nguyen, and Maxime Plancon. Lattice-based zero-knowledge proofs and applications: Shorter, simpler, and more general. *Cryptology ePrint Archive*, Paper 2022/284, 2022. <https://eprint.iacr.org/2022/284>.

- 
- [LNS20] Vadim Lyubashevsky, Ngoc Khanh Nguyen, and Gregor Seiler. Practical lattice-based zero-knowledge proofs for integer relations. *Cryptology ePrint Archive*, Paper 2020/1183, 2020. <https://eprint.iacr.org/2020/1183>.
- [LNS21] Vadim Lyubashevsky, Ngoc Khanh Nguyen, and Gregor Seiler. Shorter lattice-based zero-knowledge proofs via one-time commitments. In Juan A. Garay, editor, *Public-Key Cryptography – PKC 2021*, pages 215–241, Cham, 2021. Springer International Publishing.
- [LNSW13] San Ling, Khoa Nguyen, Damien Stehlé, and Huaxiong Wang. Improved zero-knowledge proofs of knowledge for the isis problem, and applications. In Kaoru Kurosawa and Goichiro Hanaoka, editors, *Public-Key Cryptography – PKC 2013*, pages 107–124, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.
- [Lyu12] Vadim Lyubashevsky. Lattice signatures without trapdoors. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology – EUROCRYPT 2012*, pages 738–755, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.
- [Ste94] Jacques Stern. A new identification scheme based on syndrome decoding. In Douglas R. Stinson, editor, *Advances in Cryptology — CRYPTO’ 93*, pages 13–21, Berlin, Heidelberg, 1994. Springer Berlin Heidelberg.