Grand Danois: Succinct Multilinear Polynomial Commitments over Lattices

Anders Kallesøe[1] and Hamidreza Khoshakhlagh [⋆1]

Aarhus University, Denmark
{ak, hamidreza }@cs.au.dk

Abstract. We present Grand Danois, a new post-quantum multilinear polynomial commitment scheme from lattices for polynomials over $\mathbb{F}_q$ that achieves polylogarithmic $O(\lambda\ell)$ verification complexity and proof sizes. We build on the general approach introduced in Hachi (ePrint 2026/156) with three key changes. First, we switch to the vanishing Short Integer Solution (vSIS) assumption to obtain structured public parameters for our commitment scheme and utilize this structure to design a sumcheck protocol amenable to succinct verification. Second, rather than casting ring relations into $\mathbb{F}_{q^k}[X]$, via the residual technique of Hachi, we express multiplication by fixed $\mathcal{R}_q$ elements through its rotation matrix, which lets the verifier fold each row of the constraint matrix in time linear rather than quadratic in the ring degree d. Third, we modify the quadratic relation used in Hachi and Greyhound (CRYPTO 2024) so that it becomes compatible with proving norm bounds using Johnson-Lindenstrauss projections. This is achieved through an adaptation of the structured projection strategy introduced in RoK and Roll (ASIACRYPT 2025). This has the benefit for communication complexity in that proving norm bounds and correct polynomial evaluation are integrated into a single protocol, reducing the number of commitments sent by the prover. Furthermore, we impose additional structure on our random projections to reduce the witness size even more aggressively during each round of recursion without sacrificing verification complexity. Under the vSIS assumption, our construction yields an estimated proof size of roughly 80 KB for 2^{32}-size polynomial evaluations.

1 Introduction

Multilinear polynomial commitment schemes (PCS) enable a prover to succinctly commit to an ℓ-variate multilinear polynomial $f \in \mathbb{F}_q^{\leq 1}[X_1, \ldots, X_\ell]$ and subsequently give an argument of knowledge demonstrating its evaluation $y = f(x_1, \ldots, x_\ell) \in \mathbb{F}_q$ at a public evaluation point $(x_1, \ldots, x_\ell) \in \mathbb{F}_q^\ell$. The development of multilinear PCSs has fundamentally advanced the architecture of Succinct Non-interactive ARguments of Knowledge (SNARKs). By leveraging multilinear polynomial interactive oracle proofs (PIOPs) and sumcheck protocols, modern proof systems achieve strictly linear prover complexities, circum-

⋆ At the time of this work, the author was also affiliated with Partisia.

venting the super-linear FFT overhead associated with evaluations over large multiplicative subgroups.

Despite these asymptotic improvements, the vast majority of practically deployed multilinear PCSs [KZG10, BBB+18, WTs+18, Lee21] rely on cryptographic assumptions such as the discrete logarithm (DL) problem or bilinear pairings over elliptic curves which are vulnerable to quantum algorithms. Our goal is to build a post-quantum PCS based on lattice assumptions. Such a PCS can then serve as a compiler to transform a PIOP into a quantum-safe SNARK.

Adapting multilinear PCS architectures to the lattice setting comes with some algebraic challenges. In the lattice setting, the underlying algebraic structure shifts from a finite field to a cyclotomic ring $\mathcal{R}_q = \mathbb{Z}_q[X]/(X^d+1)$ (or a related ring). Thus, for an ℓ-variate multilinear polynomial f over $\mathcal{R}_q$, the commitment and opening at an evaluation point $(x_1, \ldots, x_\ell) \in \mathcal{R}_q^\ell$ must be mapped into the ring structure. Furthermore, lattice cryptography relies heavily on bounded noise and short vectors for both security and extraction. Every inner product or folding step inherently causes norm growth. Moreover, when sumcheck protocols are applicable for reducing large polynomial relations to low-degree consistency checks, executing a standard sumcheck directly over (large) polynomial rings introduces a significant computational bottleneck for the verifier, who is required to perform NTT operations each costing $O(d \log d)$. Consequently, an efficient lattice PCS must carefully balance witness compression and strict norm bounds while ensuring an efficient verifier.

1.1 Prior Works: Polynomial Commitments with Square-Root Verification Complexity

A common approach to constructing polynomial commitments over a ring $\mathcal{R}_q$ uses the following formulation of a polynomial evaluation. Let $\ell = \ell_1 + \ell_2$ and $(f(l))_{l \in \{0,1\}^\ell}$ denote the hypercube evaluations of f defined over $\mathcal{R}_q$. We have that

$$f(X_1, \ldots, X_\ell) := \sum_{i \in \{0,1\}^{\ell_2}} \sum_{j \in \{0,1\}^{\ell_1}} f(i \| j) \cdot \mathsf{eq}\big((X_1, \ldots, X_{\ell_2}), (i_1, \ldots, i_{\ell_2})\big) \cdot \mathsf{eq}\big((X_{\ell_2+1}, \ldots, X_\ell), (j_1, \ldots, j_{\ell_1})\big)$$

where $\mathsf{eq}(\mathbf{X}, \mathbf{Y}) := \prod \big(X_i \cdot Y_i + (1 - X_i) \cdot (1 - Y_i)\big)$. The evaluation of f at $\mathbf{x} := (x_1, \ldots, x_\ell) \in \mathcal{R}_q^\ell$ can be written as the bilinear form

$$f(x_1, \ldots, x_\ell) = \mathbf{a}^\top \left[\mathbf{f}_{0\ldots0} \mid \cdots \mid \mathbf{f}_{1\ldots1}\right] \mathbf{b},$$

where the columns of the matrix $\mathbf{f} := \left[\mathbf{f}_{0\ldots0} \mid \cdots \mid \mathbf{f}_{1\ldots1}\right] \in \mathcal{R}_q^{L_1 \times L_2}$ are defined as $\mathbf{f}_i := (f(i \,\|\, j))_{j \in \{0,1\}^{\ell_1}} \in \mathcal{R}_q^{L_1}$ for $i \in \{0,1\}^{\ell_2}$, and $\mathbf{a}, \mathbf{b}$ are defined as:

$$\mathbf{a} := \Big(\mathsf{eq}\big((x_{\ell_2+1}, \ldots, x_\ell), (j_1, \ldots, j_{\ell_1})\big)\Big)_{j \in \{0,1\}^{\ell_1}} \in \mathcal{R}_q^{L_1}$$

and

$$\mathbf{b} := \Big(\mathsf{eq}\big((x_1, \ldots, x_{\ell_2}), (i_1, \ldots, i_{\ell_2})\big)\Big)_{i \in \{0,1\}^{\ell_2}} \in \mathcal{R}_q^{L_2},$$

where $L_1 = 2^{\ell_1}$, $L_2 = 2^{\ell_2}$ and $L = 2^{\ell} = L_1 L_2$.

1.1.1 Commit phase. In the lattice setting we require a norm bound for an opening of a commitment to be valid. This requires our representation of the polynomial f to adhere to this norm bound, which is of course not the case for every polynomial we might be interested in. This is commonly mediated by computing and committing to the decomposed columns of f defined as

$$\hat{\mathbf{f}}_i := \mathbf{G}_{L_1}^{-1}(\mathbf{f}_i) \in \mathcal{R}_q^{L_1 \delta}, \quad \text{where } \delta = \lceil \log q \rceil,$$

where $\mathbf{G}_{L_1}^{-1}$ computes the decomposition of each coefficient of each ring element of $\mathbf{f}_i$ modulo a sufficiently small base.

Once the polynomial coefficients are formatted as a short matrix $\hat{\mathbf{f}}$, a two-layer commitment to the polynomial is computed. Formally, let the public commitment key consist of two uniformly random matrices $\mathbf{A}$ and $\mathbf{B}$ over $\mathcal{R}_q$ with n_A and n_B rows respectively, where n_A and n_B are security parameters. The commitment to the polynomial is computed as follows:

- Commit to each $\hat{\mathbf{f}}_i$ using Ajtai commitments $\mathbf{t}_i := \mathbf{A}\hat{\mathbf{f}}_i \in \mathcal{R}_q^{n_A}$ for $i \in [L_2]$.
- Decompose each $\mathbf{t}_i$ into $\hat{\mathbf{t}}_i := \mathbf{G}_{n_A}^{-1}(\mathbf{t}_i)$ and commit to them again using another Ajtai commitment

$$\mathbf{cm} := \mathbf{B} \begin{bmatrix} \hat{\mathbf{t}}_1 \\ \vdots \\ \hat{\mathbf{t}}_{L_2} \end{bmatrix} \in \mathcal{R}_q^{n_B}.$$

We define $\mathbf{cm}$ to be the commitment to f with opening given by the tuple $(\hat{\mathbf{f}}_i, \hat{\mathbf{t}}_i)_i$.

1.1.2 Polynomial evaluation. To construct a succinct proof system for proving the evaluation $y = f(x_1, \ldots, x_\ell)$, the prover computes and sends $\mathbf{w} = (w_1, \ldots, w_{L_2}) \in \mathcal{R}_q^{L_2}$ defined as $w_i := \mathbf{a}^\top \mathbf{G}_{L_1} \hat{\mathbf{f}}_i \in \mathcal{R}_q$. Next, the verifier responds with a vector $\mathbf{c} = (c_1, \ldots, c_{L_2}) \leftarrow\!\!\$\ \mathcal{C}^{L_2}$ for a challenge set $\mathcal{C} \subset \mathcal{R}_q$ with norm bounded elements. The prover now computes $\mathbf{z} = c_1 \hat{\mathbf{f}}_1 + \cdots + c_{L_2} \hat{\mathbf{f}}_{L_2}$ over $\mathcal{R}_q$ and the verifier checks that

$$\mathbf{b}^\top \mathbf{w} = y \qquad \wedge \qquad \mathbf{A}\mathbf{z} = c_1 \mathbf{t}_1 + \cdots + c_{L_2} \mathbf{t}_{L_2}.$$

To further reduce the communication complexity of the protocol, [NS24] let the prover commit to $\mathbf{w}$ instead of sending it in the clear. The opening protocol between the prover $\mathcal{P}$ and verifier $\mathcal{V}$ proceeds as follows:

– $\mathcal{P}$ decomposes each w_i into $\hat{w}_i := \mathbf{G}_1^{-1}(w_i) \in \mathcal{R}_q^{\delta}$ and commits to them using another public uniformly random matrix $\mathbf{D}$ over $\mathcal{R}_q$ with n_D rows:

$$\mathbf{v} := \mathbf{D}\hat{\mathbf{w}} \in \mathcal{R}_q^{n_D}.$$

– $\mathcal{P}$ sends $\mathbf{v}$ to the verifier $\mathcal{V}$ and receives the challenge vector $\mathbf{c} \leftarrow\!\$\, \mathcal{C}^{L_2}$.
– $\mathcal{P}$ computes the folded witness

$$\mathbf{z} := c_1\hat{\mathbf{f}}_1 + \cdots + c_{L_2}\hat{\mathbf{f}}_{L_2} \in \mathcal{R}_q^{\delta L_1}.$$

Let $\hat{q}$ be the maximum L_∞ norm of $\mathbf{z}$, and $\hat{\delta} = \lceil \log_b \hat{q} \rceil$ for a base $b \geq 2$. Define

$$\hat{\mathbf{G}}_\eta := \mathbf{I}_\eta \otimes \left[1\ b\ b^2 \cdots b^{\hat{\delta}-1}\right] \in \mathcal{R}_q^{\eta \times \eta\hat{\delta}}.$$

To decrease the norm of $\mathbf{z}$ further, $\mathcal{P}$ uses $\hat{\mathbf{G}}_{\delta L_1}^{-1}$ to decompose $\mathbf{z}$ into $\hat{\mathbf{z}} = \hat{\mathbf{G}}_{\delta L_1}^{-1}(\mathbf{z}) \in \mathcal{R}_q^{\delta\hat{\delta}L_1}$.
– $\mathcal{P}$ proves knowledge of short vector $(\hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$ such that

$$\begin{bmatrix} \mathbf{D} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{B} & \mathbf{0} \\ \mathbf{b}^\top\mathbf{G}_{L_2} & \mathbf{0} & \mathbf{0} \\ \mathbf{c}^\top\mathbf{G}_{L_2} & \mathbf{0} & -\mathbf{a}^\top\mathbf{G}_{L_1}\hat{\mathbf{G}}_{\delta L_1} \\ \mathbf{0} & \mathbf{c}^\top \otimes \mathbf{G}_{n_A} & -\mathbf{A}\hat{\mathbf{G}}_{\delta L_1} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{w}} \\ \hat{\mathbf{t}} \\ \hat{\mathbf{z}} \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ \mathbf{cm} \\ y \\ 0 \\ \mathbf{0} \end{bmatrix}. \tag{1}$$

1.1.3 Proving linear relations in eq. (1). Directly sending $\mathbb{w} := (\hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$ to the verifier has communication complexity $O(L_1 + L_2)$ over $\mathcal{R}_q$, which is sublinear in the witness size $L = L_1L_2$. Instead of directly sending $\mathbb{w}$, Greyhound [NS24] proves eq. (1) by applying the LaBRADOR [BS23] proof system. This results in a succinct proof of size $\mathsf{poly}(\lambda, \ell)$ and a sublinear-time verifier with complexity $O(\lambda^{3/2}(L_1 + L_2))$.

1.1.4 Switching to Sumcheck over $\mathbb{F}_q$. Recent work [NOZ26] introduces a new technique that casts eq. (1) as a sumcheck-compatible relation over $\mathbb{F}_q[X]$ using a technique from [HMZ25]. As a result, the verifier does not need to perform any cyclotomic ring operations. This significantly improves concrete verifier performance and yields an asymptotic savings of a factor of $\tilde{O}(\lambda)$.

1.2 Our Contributions

In this work, we introduce Grand Danois, an efficient lattice-based multilinear polynomial commitment scheme that simultaneously achieves polylogarithmic verification time and succinct proof sizes. We achieve this through the following techniques:

- We endow the commitment matrices with a row tensor structure and develop an adaptation of the sumcheck protocol introduced in [NOZ26] that allows succinct verification. This works by extending arithmetic to $\mathcal{R}_{q^k}$.
- Rather than casting ring relations into $\mathbb{F}_{q^k}[X]$ using the residual technique of [NOZ26], we express multiplication by fixed $\mathcal{R}_q$ elements through its rotation matrix. This yields a sumcheck protocol over $\mathbb{F}_{q^k}$ in which the verifier's per-row work is linear rather than quadratic in the ring degree d.
- We improve the efficiency of proving norm bounds in two ways: (i) by ensuring compatibility with structured random projections as in [KLNO25], and (ii) by introducing additional structure on the random projections to enable more aggressive size reduction of the witness.

Our protocol builds on [NOZ26] and introduces new techniques to obtain succinct verification $O(\lambda\ell)$ and compatibility with proving norm bounds via random projections as opposed to range proofs. This comes at the cost of moving to the newer and less battle-tested vSIS assumption. We also sacrifice the property that the verifier performs no ring operations. With the goal of obtaining more aggressive size reduction of the witness at each level of recursion without compromising the verifier's complexity, we build on the structured random projection technique introduced in [KLNO25]. We also sacrifice the ability to commit to polynomials over $\mathbb{F}_{q^k}$ although we note that this could be achieved using the same embedding techniques as in [NOZ26] at the cost of concretely larger proofs.

1.3 Technical Overview

We aim to build a polynomial commitment scheme for multilinear polynomials over $\mathbb{F}_q$. Our starting point is similar to that of Greyhound [NS24] except we consider a commitment scheme whose binding is based on the vanishing SIS assumption and our polynomial is over $\mathbb{F}_q$. We consider a polynomial $f \in \mathbb{F}_q^{\leq 1}[X_1, \ldots, X_\ell]$. Evaluating $f(\mathbf{x})$ can be expressed via the following quadratic equation:

$$f(x_1, \ldots, x_\ell) = \mathbf{a}^\top [f(0 \ldots 0) | \ldots | f(1 \ldots 1)]\mathbf{b}, \tag{2}$$

where $\mathbf{a}, \mathbf{b}$ are defined similarly to section 1.1. The 2^ℓ evaluations of f can be naturally packed into a d-times-shorter collection of $\mathcal{R}_q$ elements: we write $\mathbf{s} = [\mathbf{s}_1 \mid \cdots \mid \mathbf{s}_r] \in \mathcal{R}_q^{L_1 \times r}$, where $r = L_2/d$. Correspondingly, $\mathbf{b}$ is now indexed over these r packed columns rather than over the original L_2 hypercube points, while $\mathbf{a} \in \mathbb{F}_q^{L_1}$ is unchanged. From this perspective, we can create a commitment to f by computing $\mathbf{t}_i = \mathbf{A}\hat{\mathbf{s}}_i$ as in section 1.1. We note that multiplication between $\mathbf{a}^\top$ and $\mathbf{s}$ is well defined by simply considering $\mathbf{a}^\top$ to be a scalar vector for the module $\mathcal{R}_q$. Now, we want to support evaluation claims over $\mathbb{F}_{q^k}$ (for some k) but to save on proof size we wish to avoid the embedding used in [NOZ26]. We instead consider the straightforward extension of $\mathcal{R}_q$ to $\mathcal{R}_{q^k}$. Now, we can consider $\mathbf{a}^\top$ to be a scalar vector for the module $\mathcal{R}_{q^k}$. From here, we can proceed as in section 1.1. The prover first computes $\mathbf{w}_i = \mathbf{a}^\top \mathbf{G}_{L_1} \hat{\mathbf{s}}_i$ and $\mathbf{t}_i$-s and the prover and verifier proceed as before up to

$$\begin{bmatrix} \mathbf{D} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{B} & \mathbf{0} \\ \mathbf{I}_k \otimes \mathbf{c}^\top \mathbf{G}_r & \mathbf{0} & -\mathbf{a}^\top \mathbf{G}_{L_1} \hat{\mathbf{G}}_{\delta L_1} \\ \mathbf{0} & \mathbf{c}^\top \otimes \mathbf{G}_{n_A} & -\mathbf{A}\hat{\mathbf{G}}_{\delta L_1} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{w}} \\ \hat{\mathbf{t}} \\ \hat{\mathbf{z}} \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ \mathbf{cm} \\ 0 \\ \mathbf{0} \end{bmatrix},$$

with the key difference that verifying that $\mathbf{a}^\top$ was applied correctly now spans k constraints. Furthermore, the verifier should now check that

$$\langle \mathbf{b}, \mathbf{w} \rangle = y$$

holds over $\mathbb{F}_{q^k}$. Next, we want to recursively prove knowledge of short $(\hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$ that satisfy the above constraints. We follow the approach of Hachi with some modifications that aim to achieve an even more succinct verifier. Firstly, there is the norm check. In Hachi, this is done with respect to the l_∞ norm via a range check on the coefficients of $\mathbb{w} = (\hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$. Proving that $\mathbb{w}$ is short is sufficient to extract weak openings for the $\mathbf{t}_i$. However, the communication complexity of this check scales linearly with the size of the decomposition parameter. A popular alternative that was first introduced in LaBRADOR [BS23] is via random Johnson-Lindenstrauss projections. These mappings can be shown to maintain the l_2 norm of vectors up to a small bias with overwhelming probability, despite the projection dimension being very small. Since the projection is random and has the same size as the witness, this inevitably creates a linear verifier. RoK and Roll [KLNO25] introduced an approach to impose structure on these projections in order to make them amenable to succinct verification. This is done by sampling a random projection that is small in length and then applying it to the witness as a convolution, i.e., to limbs of the witness in parallel. This comes at the cost of making the dimension of the projection linear in the length of the witness. To handle this, the authors design a protocol that commits to the projection and proves shortness recursively while ensuring that a short projection can be extracted by the knowledge extractor.

In this work, we follow the same approach to projecting the witness except for two differences: we impose further structure on the projection, and we integrate the projection step into the above constraint system to simultaneously prove correctness of an evaluation of f and shortness of $\mathbf{s}$ via its projection. The structure that we impose on the projection is sampling two smaller, independent projections $\mathbf{J}, \mathbf{J}' : \mathcal{R}_q^{256j} \to \mathcal{R}_q^{256}$ and using the projection $(\mathbf{I}_r \otimes \mathbf{J}')(\mathbf{I}_{rj} \otimes \mathbf{J})$ where $j^2 = L_2/d = r$. This allows a more aggressive reduction in the witness size at each step of recursion without the need for the verifier to sample and manipulate too much randomness. Furthermore, it does not introduce too large of a bias when viewing the l_2 norm of the projection as an estimator for the l_2 norm of f. What we then note is that the application of this projection map to f fits into the above framework. We simply consider only the first step. That is, $(\mathbf{I}_r \otimes \mathbf{J}')(\mathbf{I}_{rj} \otimes \mathbf{J})\mathbf{s}$ can be seen as applying a smaller projection map $\hat{\mathbf{J}}$ to each $\mathbf{s}_i$, i.e., the computation $\hat{\mathbf{J}}[\mathbf{s}_1 \mid \cdots \mid \mathbf{s}_r]$. Let the resulting projection be $\mathbf{p}$. Verifying that $\mathbf{p}$ is the correct result of applying the projection to $\mathbf{s}$ can be done probabilistically by checking that $\hat{\mathbf{J}}\mathbf{z} = \mathbf{c}^\top \mathbf{p}$. Integrating this into the above constraints gives:

$$\begin{bmatrix} \mathbf{D}_1 & \mathbf{D}_2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{B} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_k \otimes (\mathbf{c}^\top \mathbf{G}_r) & \mathbf{0} & -\mathbf{a}^\top \mathbf{G}_{L_1} \hat{\mathbf{G}}_{\delta L_1} \\ (\mathbf{c}^\top \mathbf{G}_{L_1}) & \mathbf{0} & \mathbf{0} & -\hat{\mathbf{J}} \hat{\mathbf{G}}_{\delta L_1} \\ \mathbf{0} & \mathbf{0} & \mathbf{c}^\top \otimes \mathbf{G}_{n_A} & -\mathbf{A} \hat{\mathbf{G}}_{\delta L_1} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{p}} \\ \hat{\mathbf{w}} \\ \hat{\mathbf{t}} \\ \hat{\mathbf{z}} \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ \mathbf{cm} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \tag{3}$$

Here, we commit to $(\hat{\mathbf{p}}, \hat{\mathbf{w}})$ in a single commitment, where $\mathbf{D}_1$ denotes the part of $\mathbf{D}$ used for $\hat{\mathbf{p}}$ and $\mathbf{D}_2$ that for $\hat{\mathbf{w}}$. Proving knowledge of a small $(\hat{\mathbf{p}}, \hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$ that satisfies the above linear equations now convinces the verifier that $f(x_1, \ldots, x_\ell) = y$ and that f has a small l_2 norm.

Next, we consider how the prover would prove knowledge of such a $(\hat{\mathbf{p}}, \hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$. In Hachi [NOZ26], the authors consider the general case of an unstructured matrix-vector product over $\mathcal{R}_q$:

$$\mathbf{M}\mathbf{x} = \mathbf{y}$$

They use the observation that by considering the entries of $\mathbf{M}, \mathbf{x}, \mathbf{y}$ as polynomials over $\mathbb{F}_q$, the above equation implies that there exists a vector $\mathbf{r}$ over $\mathbb{F}_q[X]^{<d}$ such that

$$\mathbf{M}\mathbf{x} = \mathbf{y} + (X^d + 1)\mathbf{r}, \tag{4}$$

where the computation is taken over $\mathbb{F}_q[X]$. From here, they construct a sumcheck protocol over the multilinear extension of $(\mathbf{x}, \mathbf{r})$ to prove that the above equation holds. We are also interested in casting the verification as a sumcheck protocol over a small extension of $\mathbb{F}_q$, but rather than considering $\mathcal{R}_q$ elements as general polynomials, we instead turn to the rotation matrices that formulate multiplication in $\mathcal{R}_q$. For the operation

$$\begin{aligned} \mathsf{rot} : \mathbb{F}_q^d &\to \mathbb{F}_q^d \\ \mathsf{rot}(a_0, \ldots, a_{d-1}) &= (-a_{d-1}, a_0, \ldots, a_{d-2}), \end{aligned}$$

multiplication by $a \in \mathcal{R}_q$ can be expressed via the matrix

$$\mathbf{M}_a := [\mathsf{cf}(a), \mathsf{rot}(\mathsf{cf}(a)), \mathsf{rot}^2(\mathsf{cf}(a)), \ldots, \mathsf{rot}^{d-1}(\mathsf{cf}(a))] \in \mathbb{F}_q^{d \times d}, \tag{5}$$

which operates on coefficient vectors of $\mathcal{R}_q$ elements. We wish to utilize this expression of multiplication by ring elements to construct an efficient sumcheck protocol by which to verify linear relations over $\mathcal{R}_q$. As a starting point, we can consider a single multiplication of the elements $a, b \in \mathcal{R}_q$

$$ab = c \tag{6}$$

Expressing this via vectors and matrices over $\mathbb{F}_q$ we get

$$\mathbf{M}_a \mathsf{cf}(b) = \mathsf{cf}(c) \tag{7}$$

There is a well-known, sumcheck-compatible approach to verifying such a matrix-vector multiplication. Namely, if we wanted to verify $\mathbf{Mx} = \mathbf{y}$, we can check if $\boldsymbol{\alpha}^\top \mathbf{Mx} = \boldsymbol{\alpha}^\top \mathbf{y}$ where $\boldsymbol{\alpha} = (1, \alpha, \alpha^2, \ldots, \alpha^{d-1})$ for a random α. This could be expressed as the sum

$$\sum_{a,b \in \{0,1\}^{\log d}} \tilde{\boldsymbol{\alpha}}(a) \widetilde{\mathbf{M}}(a,b) \tilde{\mathbf{x}}(b) \tag{8}$$

which could be immediately be checked in an invocation of the sumcheck protocol over the multilinear extensions of $\boldsymbol{\alpha}, \mathbf{M}, \mathbf{x}$. An apparent downside here is the quadratic computational complexity in d. We wish to improve on that. Firstly, we note that multiplication by $\boldsymbol{\alpha}$ need not be expressed as a separate polynomial in the sumcheck protocol. Rather, we can instead consider the multilinear extension of the folded row vector $\boldsymbol{\alpha}\mathbf{M}$, so that the sum collapses to

$$\sum_{b \in \{0,1\}^{\log d}} \widetilde{(\boldsymbol{\alpha}\mathbf{M})}(b)\, \tilde{\mathbf{x}}(b). \tag{9}$$

This still does not solve the verifier's quadratic complexity in d. However, for the specific type of matrix that we are interested in, we can in fact do better for the verifier. Namely, the vector $\boldsymbol{\alpha}\mathbf{M}_a$ can be computed in time $O(d)$. Firstly, compute the inner product $\boldsymbol{\alpha}\mathsf{cf}(a)$. Next, $\boldsymbol{\alpha}\mathsf{rot}(\mathsf{cf}(a))$ can be computed as $(\boldsymbol{\alpha}\mathsf{cf}(a) - a_{d-1}\alpha^{d-1})\alpha - a_{d-1}$, and the same approach can be applied to compute the rest of the entries. From there, an evaluation of $\widetilde{\boldsymbol{\alpha}\mathbf{M}}$ can be computed directly in $O(d)$ time. Next, we consider how to use this observation to construct an efficient interactive proof for the relation that we are interested in, a matrix-vector product over $\mathcal{R}_q$:

$$\mathbf{Mx} = \mathbf{y}$$

Following the same technique as above for each entry in $\mathbf{M}$ we can define $\mathbf{M}_\alpha$. This is the matrix

$$\begin{bmatrix} \boldsymbol{\alpha}\mathbf{M}_{1,1} & \boldsymbol{\alpha}\mathbf{M}_{1,2} & \ldots & \boldsymbol{\alpha}\mathbf{M}_{1,m} \\ \vdots & & & \\ \boldsymbol{\alpha}\mathbf{M}_{n,1} & \boldsymbol{\alpha}\mathbf{M}_{n,2} & \ldots & \boldsymbol{\alpha}\mathbf{M}_{n,m} \end{bmatrix} \tag{10}$$

Where each $\mathbf{M}_{i,j}$ is the multiplication matrix corresponding to entry $\{i,j\}$ of $\mathbf{M}$. The sumcheck relation we get is now

$$\sum_{a \in \{0,1\}^{\log d}, b \in \{0,1\}^{\log md}} \boldsymbol{\beta}(a) \mathbf{M}_\alpha(a,b) \mathbf{x}(b)$$

Where $\boldsymbol{\beta}$ is constructed similarly to $\boldsymbol{\alpha}$, i.e., $\boldsymbol{\beta} := (1, \beta, \beta^2, \ldots, \beta^{n-1})$ for a random $\beta \in \mathbb{F}_{q^k}$. Once again, we could bake the folding by $\boldsymbol{\beta}$ of $\mathbf{M}_\alpha$ into a single polynomial ${}_\beta\mathbf{M}_\alpha$ and consider the sum

$$\sum_{b \in \{0,1\}^{\log md}} \widetilde{{}_\beta\mathbf{M}_\alpha}(b) \tilde{\mathbf{x}}(b) \tag{11}$$

In our protocol we specifically consider the vanishing SIS assumption. This is to obtain structure in $\mathbf{M}$ that can be utilized by the verifier to efficiently evaluate the multilinear extension of ${}_{\beta}\mathbf{M}_{\alpha}$. Namely, for vSIS public parameter matrix $\mathbf{M}$ the evaluation $\widetilde{{}_{\beta}\mathbf{M}_{\alpha}}(e)$ can be expressed as

$$(\boldsymbol{\beta} \otimes \boldsymbol{\alpha}) \begin{bmatrix} \prod_{i=1}^{\log m} (\mathbf{I}_d(1-e_i) + \mathbf{M}_{1,i} e_i) \\ \vdots \\ \prod_{i=1}^{\log m} (\mathbf{I}_d(1-e_i) + \mathbf{M}_{n,i} e_i) \end{bmatrix} \mathbf{e}'$$

where $\mathbf{e}'$ is a multilinear evaluation vector corresponding to the remaining $\log d$ variables in e. The observation that we will utilize is that each entry in the column vector of the above expression can be viewed as an element of $\mathcal{R}_{q^k}$. Thus, each entry can be computed using $\log n$ multiplications over $\mathcal{R}_{q^k}$.

We have yet to ensure that it can also be done for $\hat{\mathbf{J}}$ and $\mathbf{a}^\top$. We start with $\hat{\mathbf{J}}$ which has a diagonal block structure with block matrix $\mathbf{J}'(\mathbf{I}_\jmath \otimes \mathbf{J})$. This is a $256 \times 256\jmath^2$ matrix that we implicitly embed into $\mathcal{R}_q$ as constant polynomials. We wish to compute a partial evaluation of a row of $(\mathbf{J}'(\mathbf{I}_\jmath \otimes \mathbf{J}))_\alpha$ at a point e over $\mathbb{F}_{q^k}$. This involves taking the inner product between the evaluation vector corresponding to e and the given row of $(\mathbf{J}'(\mathbf{I}_\jmath \otimes \mathbf{J}))$. Since e is tensor structured we can split it into $e' \otimes e''$, where e' matches the width of $\mathbf{J}$. Now, we can compute $\mathbf{J}e'$ which will at most be a polylogarithmic-time computation. Next, we compute $\mathbf{J}'((\mathbf{J}e') \otimes e'')$ which also takes at most poly-logarithmic time. The resulting vector corresponds to the partial evaluation of each row of $(\mathbf{J}'(\mathbf{I}_\jmath \otimes \mathbf{J}))$. Notably, each of these results is a constant polynomial over $\mathcal{R}_{q^k}$.

The vector $\mathbf{a}^\top$ requires some care. We cannot simply treat its coefficients as constant polynomials in $\mathcal{R}_{q^k}$ whose multiplication matrices are simply scaled identity matrices. This is because in our constraint system we consider $\mathbf{a}$ to be a matrix with k rows rather than a single vector over $\mathbb{F}_{q^k}$. However, by again considering the matrix form of the linear operation of multiplying by a given element of $\mathbb{F}_{q^k}$, we obtain a tensor structured computation that we can perform succinctly. Namely, after computing the necessary combination of such matrices, the first column of the result can now be treated as k constant polynomials in $\mathcal{R}_{q^k}$ which can now be treated in line with our previous strategy.

1.4 Related Work

Historically, polynomial commitment schemes have been dominated by group-based and hash-based paradigms. Group-based schemes such as KZG [KZG10] and Bulletproofs IPA [BBB+18] achieve excellent concrete efficiency, but they rely on assumptions that are vulnerable to quantum adversaries. In contrast, hash-based schemes such as FRI [BBHR18] and Brakedown [GLS+23] offer post-quantum security and transparent setup; however, the absence of algebraic structure limits the extent to which succinctness and efficiency can be improved. This tension has motivated the study of lattice-based polynomial commitments which aim to combine post-quantum security with richer algebraic structure.

Lattice-based cryptography has seen substantial advances in recent years, driven by the post-quantum hardness of SIS and LWE. Its rich algebraic structure has enabled both advanced functionality and efficient protocol design, as illustrated by recent succinct proof systems stemming from LaBRADOR [BS23]. LaBRADOR achieves significantly smaller proof sizes than previous works through two key techniques: approximate range proofs based on JL transforms and recursively proving knowledge of correct openings to nested commitments satisfying the required verifier predicates. The latter is made possible by the fact that in Ajtai commitments, the value space and commitment space are the same. Thus, algebraic statements about commitment values can be expressed directly as constraints over value vectors in the commitment scheme. However, the LaBRADOR verifier still has linear complexity.

Greyhound [NS24] improves this to square-root complexity by exploiting the tensor structure of polynomial evaluation statements and verifying them through an interactive protocol. To further reduce verification complexity to sublinear, Hachi [NOZ26] combines the tensor relations introduced in Greyhound with a sumcheck protocol over cyclotomic rings. By incorporating a ring-switching technique, it lifts statements over $\mathcal{R}_q$ to a polynomial ring, thereby reducing the verifier's workload.

Recently, a line of works [BC25a, BC25b, NS25, NS26] introduced lattice-based folding schemes that can serve as building blocks for incrementally verifiable computation (IVC) [Val08] and proof-carrying data (PCD) [CT10, BCCT13]. Similar to our approach, SuperNeo [NS26]—an extension of Neo—utilizes a scalar extension of the cyclotomic ring $\mathcal{R}_q$. Specifically, Neo originally proposed a technique for obtaining commitments to vectors over $\mathbb{F}_q$ using Ajtai commitments while preserving an evaluation homomorphism over $\mathbb{F}_q$. In SuperNeo [NS26] this same approach is extended in such a manner that it also holds for $\mathbb{F}_{q^k}$ such that a constraint system can be proven via sumcheck over $\mathbb{F}_{q^k}$. This is achieved by extending the arithmetic to $\mathcal{R}_{q^k}$.

2 Preliminaries

2.1 Algebraic Number Theory

Let q be an odd prime and let d be a power of two. The $2d$-th cyclotomic ring is $\mathcal{R} = \mathbb{Z}[X]/(X^d + 1)$. Throughout the paper we work modulo q, and denote $\mathcal{R}_q = \mathbb{Z}_q[X]/(X^d + 1)$. We use $\mathbb{Z}_q$ and $\mathbb{F}_q$ interchangeably throughout the paper. We use lower-case letters for elements in $\mathcal{R}$ or $\mathcal{R}_q$ and bold lower-case (resp. upper-case) letters for representing column vectors (resp. matrices) with coefficients in $\mathcal{R}$ or $\mathcal{R}_q$. For $y = \sum_{i=0}^{d-1} y_i X^i \in \mathcal{R}_q$, we denote its coefficient vector by $\mathsf{cf}(y) := (y_0, \ldots, y_{d-1}) \in \mathbb{Z}_q^d$ and its constant coefficient by $\mathsf{ct}(y) = y_0$.

We consider base-b gadget matrices $\mathbf{G}_{b,\eta} := \mathbf{I}_\eta \otimes \left[1\ b\ b^2 \cdots b^{\delta-1}\right] \in \mathcal{R}_q^{\eta \times \eta\delta}$ for $\eta \geq 1$ and $\delta = \lceil \log_b q \rceil$. We define the inverse function $\mathbf{G}_{b,\eta}^{-1} : \mathcal{R}_q^\eta \to \mathcal{R}_q^{\eta\delta}$ which decomposes entries with respect to base b, so that for $\mathbf{t} \in \mathcal{R}_q^\eta$, $\mathbf{G}_{b,\eta}^{-1}(\mathbf{t})$ has coefficients between $\lceil -b/2 \rceil$ and $\lceil b/2 \rceil - 1$, and $\mathbf{G}_{b,\eta}\mathbf{G}_{b,\eta}^{-1}(\mathbf{t}) = \mathbf{t}$. We may omit

Symbol	Description
λ	security parameter
q	prime modulus
d	ring degree, a power of two, $d = d(\lambda)$
$\mathcal{R}$	$\mathbb{Z}[X]/(X^d + 1)$
$\mathcal{R}_q$	$\mathbb{Z}_q[X]/(X^d + 1)$
k	extension degree, $k = \lambda / \log q$
$\mathbb{F}_{q^k}, \mathcal{R}_{q^k}$	extension field / ring
b	decomposition base
δ	$\delta = \lceil \log_b q \rceil$
$\mathbf{G}_{b,\eta}$	base-b gadget matrix $\mathbf{G}_{b,\eta} := \mathbf{I}_\eta \otimes \left[1\ b\ b^2 \cdots b^{\delta-1}\right]$
$\ell = \ell_1 + \ell_2$	number of polynomial variables
$L_i = 2^{\ell_i}$; $L = 2^\ell = L_1 L_2$	hypercube sizes
$r = L_2/d$	#columns of the packed witness
$\jmath$	splitting factor, $\jmath^2 = L_2/d = r$
$\mathcal{C}$	challenge set, $\mathcal{C} \subset \mathcal{R}_q$
n_A, n_B, n_D	row counts of the CRS matrices, all $O(1)$
$\mathbf{A}, \mathbf{B}, \mathbf{D} = [\mathbf{D}_1 \mid \mathbf{D}_2]$	public CRS matrices
$\mathbf{J}, \mathbf{J}' \in \{-1, 0, 1\}^{256 \times 256\jmath}$	structured random projections
$\hat{\mathbf{J}} = \mathbf{J}'(\mathbf{I}_\jmath \otimes \mathbf{J})$	$256 \times 256\jmath^2$ composed projection
rot	coefficient rotation, section 3.2.1
$\mathsf{MSIS}_{q,d,n,m,\beta}$	Module-SIS assumption
$\mathsf{vSIS}_{q,d,n,\beta}$	vanishing-SIS assumption
$\mathcal{X}_{n,\mu}$	vSIS-friendly tensor distribution over $\mathcal{R}_q^{n \times 2^\mu}$

Table 1: Notation used throughout the paper.

the subscript b if the base is understood from context or if the explicit base is not important and instead distinguish possibly different bases via accents such as $\bar{\cdot}$.

We write $x \leftarrow \mathcal{D}$ to denote that x is sampled from the distribution $\mathcal{D}$. For sampling uniformly at random from a finite set S, we write $x \leftarrow\!\!\$\ S$. For a positive integer n, we denote $[n] = \{1, \ldots, n\}$.

For vectors over $\mathbb{Z}_q$, we consider their norms in terms of their centered integer representatives, i.e., as vectors in $[-(q-1)/2, (q-1)/2]$. We take the infinity norm of a vector $\mathbf{w}$ over $\mathbb{Z}_q$ to be $\|\mathbf{w}\|_\infty = \max_i |w_i|$ and we take the l_2 norm to be $\|\mathbf{w}\|_2 = \sqrt{w_1^2 + \cdots + w_{|\mathbf{w}|}^2}$. For a vector $\mathbf{v}$ over $\mathcal{R}_q$, we consider $\|\mathbf{v}\|_\infty, \|\mathbf{v}\|_2$ over the concatenation of the coefficients of each of the entries in $\mathbf{v}$. For ease of reference, a summary of the notation is provided in table 1.

2.2 Polynomials and Multilinear Extensions

Let $\mathbf{X} = (X_0, \ldots, X_{\mu-1})$ and $\mathbf{Y} = (Y_0, \ldots, Y_{\mu-1})$ denote vectors of formal variables. We let $\mathcal{R}_q[\mathbf{X}]$ denote the set of multivariate polynomials over $\mathcal{R}_q$ in the variables $\mathbf{X}$. For any vector $\mathbf{w} \in \mathcal{R}_q^{2^\mu}$, we let $\mathsf{MLE}[\mathbf{w}](\mathbf{X}) \in \mathcal{R}_q[\mathbf{X}]$ denote its

unique multilinear extension. Specifically, $\mathbf{w}$ is treated as the evaluation table of a function $\{0,1\}^\mu \to \mathcal{R}_q$, where the domain elements $\{0,1\}^\mu$ are mapped to indices in $[2^\mu]$ via a standard decomposition. For $\mathbf{x} = (x_0, x_1, \ldots, x_{\mu-1}) \in \mathcal{R}_q^\mu$, by definition it holds that

$$\mathsf{MLE}[\mathbf{w}](\mathbf{x}) = ((1-x_0, x_0) \otimes \ldots \otimes (1-x_{\mu-1}, x_{\mu-1}))^\top \cdot \mathbf{w}.$$

We define the multilinear equality polynomial $\mathsf{eq} \in \mathcal{R}_q[\mathbf{X}, \mathbf{Y}]$ as

$$\mathsf{eq}(\mathbf{X}, \mathbf{Y}) := \prod_{i=0}^{\mu-1} \big(X_i \cdot Y_i + (1-X_i) \cdot (1-Y_i)\big).$$

Over the Boolean hypercube $\mathbf{X}, \mathbf{Y} \in \{0,1\}^\mu$, this polynomial acts as an exact indicator function yielding $\mathsf{eq}(\mathbf{X}, \mathbf{Y}) = 1$ if $\mathbf{X} = \mathbf{Y}$, and 0 otherwise.

2.3 Hardness Assumptions

Our construction relies on two hardness assumptions over lattices: the standard Module Short Integer Solution (MSIS) problem, and its structured variant, the Vanishing Short Integer Solution (vSIS) problem.

Definition 2.1 (Module-SIS). *Let $q = q(\lambda)$, $n = n(\lambda)$, $m = m(\lambda)$, $d = d(\lambda)$, and $\beta = \beta(\lambda)$ be parameters parameterized by the security parameter λ. We say that the $\mathsf{MSIS}_{q,d,n,m,\beta}$ assumption holds if for any PPT adversary $\mathcal{A}$,*

$$\Pr\left[\begin{array}{c} \boldsymbol{B}\boldsymbol{t} = \mathbf{0} \bmod q \\ \wedge \\ 0 \neq \|\boldsymbol{t}\| \leq \beta \end{array} \;\middle|\; \begin{array}{l} \boldsymbol{B} \leftarrow\!\$\, \mathcal{R}_q^{n\times m} \\ \boldsymbol{t} \leftarrow \mathcal{A}(\boldsymbol{B}) \end{array}\right] \leq \mathsf{negl}(\lambda).$$

The Vanishing Short Integer Solution (vSIS) problem is a structured variant of the standard SIS problem. The primary distinction is that the underlying matrix $\mathbf{A} \in \mathcal{R}_q^{n\times 2^\mu}$ is not sampled uniformly at random, but instead obtained via the iterated row-tensor product of smaller base matrices; i.e., $\mathbf{A} = \mathbf{A}_0 \otimes \mathbf{A}_1 \otimes \cdots \otimes \mathbf{A}_{\mu-1}$, where each $\mathbf{A}_i \in \mathcal{R}_q^{n\times 2}$. We denote by $\mathcal{R}_q^{n\times 2^{\otimes\mu}} \subset \mathcal{R}_q^{n\times 2^\mu}$ the algebraic domain of matrices with such tensor structure.

Definition 2.2 (vSIS (adapted from [CLM23])). *Let $q = q(\lambda)$, $n = n(\lambda)$, $d = d(\lambda)$, and $\beta = \beta(\lambda)$ be parameters parameterized by the security parameter λ. Let $\mathcal{X}_{n,\mu}$ be a distribution over $\mathcal{R}_q^{n\times 2^{\otimes\mu}}$. We say that the $\mathsf{vSIS}_{q,d,n,\beta}$ assumption w.r.t. $\mathcal{X}_{n,\mu}$ holds if for any PPT adversary $\mathcal{A}$,*

$$\Pr\left[\begin{array}{c} \boldsymbol{A}\boldsymbol{w} = \mathbf{0} \bmod q \\ \wedge \\ 0 \neq \|\boldsymbol{w}\| \leq \beta \end{array} \;\middle|\; \begin{array}{l} \boldsymbol{A} \leftarrow\!\$\, \mathcal{X}_{n,\mu} \\ \boldsymbol{w} \leftarrow \mathcal{A}(\boldsymbol{A}) \end{array}\right] \leq \mathsf{negl}(\lambda).$$

2.4 Interactive Proofs and Special Soundness

We define our protocols over ternary relations $\mathsf{R} \subseteq \mathbb{P} \times \mathbb{X} \times \mathbb{W}$ consisting of public parameters, statements and witnesses. We say that $\mathbb{w}$ is a valid witness for $(\mathbb{pp}, \mathbb{x})$ if $\mathsf{R}(\mathbb{pp}, \mathbb{x}, \mathbb{w}) = 1$.

An interactive proof system $\Pi = (\mathcal{G}, \mathcal{P}, \mathcal{V})$ for relation R consists of a setup algorithm $\mathcal{G}(1^\lambda)$ that generates public parameters $\mathbb{pp}$, and a pair of interactive PPT algorithms $(\mathcal{P}, \mathcal{V})$. On common input $(\mathbb{pp}, \mathbb{x})$ and with the prover additionally given a valid witness $\mathbb{w}$ for $(\mathbb{pp}, \mathbb{x})$, the prover and verifier engage in an interactive protocol at the end of which they output a transcript tr and the verifier outputs a decision bit $b \in \{0, 1\}$. We denote this execution by

$$(\mathsf{tr}, b) \leftarrow \langle \mathcal{P}(\mathbb{pp}, \mathbb{x}, \mathbb{w}), \mathcal{V}(\mathbb{pp}, \mathbb{x}) \rangle$$

All protocols in this work are public-coin, meaning the verifier's challenges in every round are chosen uniformly at random and independently of the prover's messages. Such protocols can be transformed into non-interactive proof systems using Fiat-Shamir transformation.

It was shown in [FMN23] that to prove knowledge soundness, it is sufficient to prove that the protocol satisfies a form of special soundness called *coordinate-wise special soundness* (CWSS)[1]. We therefore define only coordinate-wise special soundness and show that our protocol satisfies this property.

Let $\mathcal{C}$ be a challenge set and $\nu \in \mathbb{N}$. For any two vectors $\mathbf{u} = (u_1, \ldots, u_\nu)$ and $\mathbf{v} = (v_1, \ldots, v_\nu)$ in $\mathcal{C}^\nu$, we recall the definition of coordinate-wise difference relation $\equiv_i$ which indicates that $\mathbf{u}$ and $\mathbf{v}$ are identical in all coordinates except for the i-th coordinate. Formally, for a fixed index $i \in [\nu]$ this is defined as:

$$\mathbf{u} \equiv_i \mathbf{v} \iff u_i \neq v_i \ \wedge \ \forall j \in [\nu] \setminus \{i\},\ u_j = v_j.$$

Using this relation, we next recall the coordinate-wise special soundness challenge set $\mathsf{SS}(\mathcal{C}, \nu, \kappa)$ as the set of all tuples containing $K = \nu(\kappa - 1) + 1$ vectors where one "central" vector $\mathbf{c}_e$ differs from the rest in exactly one distinct coordinate each. In other words, for every coordinate, there are κ vectors in the set that are identical except for that coordinate, with $\mathbf{c}_e$ always included among them. Formally:

$$\mathsf{SS}(\mathcal{C}, \nu, \kappa) := \left\{ (\mathbf{c}_1, \ldots, \mathbf{c}_K) \in (\mathcal{C}^\nu)^K \;\middle|\; \begin{array}{l} \exists e \in [K] \text{ s.t. } \forall i \in [\nu], \\ \exists J = \{j_1, \ldots, j_{\kappa-1}\} \subseteq [K] \setminus \{e\} \\ \text{ s.t. } \forall j \in J : \mathbf{c}_e \equiv_i \mathbf{c}_j \end{array} \right\}.$$

We now define CWSS for multi-round interactive protocols.

Definition 2.3 ($(\nu_1, \ldots, \nu_\mu)$-Coordinate-Wise $(\kappa_1, \ldots, \kappa_\mu)$-Special Soundness). *Let $\Pi = (\mathcal{G}, \mathcal{P}, \mathcal{V})$ be a public coin $(2\mu + 1)$-message interactive proof system for a relation $\mathsf{R} \subseteq \mathbb{P} \times \mathbb{X} \times \mathbb{W}$. For each round $i \in [\mu]$, suppose the*

[1] This holds for both the interactive case and the non-interactive variant in the random oracle (see Lemma 2.31 in [FMN23]).

verifier's challenge is drawn from a set $\mathcal{C}_i^{\nu_i}$*. We say that* Π *satisfies* $(\nu_1, \ldots, \nu_\mu)$*-coordinate-wise* $(\kappa_1, \ldots, \kappa_\mu)$*-special soundness for the relation* R *if there exists a deterministic polynomial time algorithm* Ext *with the following property:*

Let $\mathcal{T}$ *be a valid tree of accepting transcripts arranged such that nodes correspond to prover messages and edges correspond to verifier challenges. The tree possesses the following structure:*

- *The total number of transcripts is exactly* $K := \prod_{i=1}^{\mu} \big(\nu_i(\kappa_i - 1) + 1\big)$*.*
- *At depth* $i \in [\mu]$ *(corresponding to the* i*-th verifier challenge), each node branches into exactly* $\nu_i(\kappa_i - 1) + 1$ *children.*
- *The challenges along these branching edges form a valid coordinate-wise special soundness set* $\mathsf{SS}(\mathcal{C}_i, \nu_i, \kappa_i)$*. All prover messages preceding this branch remain identical across these paths.*

Given public parameters $\mathbb{pp} \in \mathbb{P}$*, a statement* $\mathbb{x} \in \mathbb{X}$*, and such a transcript tree* $\mathcal{T}$*,* Ext *outputs a valid witness* $\mathbb{w}^*$ *for* $(\mathbb{pp}, \mathbb{x})$*.*

2.5 Polynomial Commitment Schemes

A Polynomial Commitment Scheme (PCS) functions as a standard commitment to a polynomial f, augmented with an interactive protocol to prove evaluations of f at specific points. Because lattice-based constructions inherently incur noise growth, the definition should be defined with a slack space SL and a fixed identity element $\mathsf{e} \in \mathsf{SL}$ representing zero slack.

Definition 2.4 (Polynomial Commitment Scheme). *A polynomial commitment scheme for* ℓ*-variate multilinear polynomials and a slack space* SL *is a tuple of algorithms* $\mathsf{PCS} = (\mathsf{Setup}, \mathsf{Commit}, \mathsf{Open}, \mathsf{Eval})$ *defined as follows.*

- $\mathsf{Setup}(1^\lambda, \ell) \to \mathbb{pp}$*: Outputs public parameters* $\mathbb{pp} \in \mathbb{P}$*.*
- $\mathsf{Commit}(\mathbb{pp}, f) \to (\mathbf{cm}, st)$*: Takes a polynomial* $f \in \mathcal{R}^{\leq 1}[X_1, \ldots, X_\ell]$ *and outputs a commitment* $\mathbf{cm}$ *and decommitment state* st*.*
- $\mathsf{Open}(\mathbb{pp}, \mathbf{cm}, f, st, \mathsf{c}) \to \{0, 1\}$*: Outputs a bit indicating whether the input* $\mathbf{cm}$ *is a valid commitment to* f *under* $\mathbb{pp}$*, state* st *and slack factor* c*.*
- $\mathsf{Eval} = \langle \mathcal{P}_{\mathsf{eval}}, \mathcal{V}_{\mathsf{eval}} \rangle$*: An interactive argument of knowledge between a prover* $\mathcal{P}_{\mathsf{eval}}$ *and verifier* $\mathcal{V}_{\mathsf{eval}}$*. The common inputs are* $(\mathbb{pp}, \mathbf{cm}, \boldsymbol{x}, y)$*, and the prover's private inputs are* (f, st)*.*

The evaluation protocol Eval must satisfy completeness and knowledge soundness. Specifically, it must be an argument of knowledge for the following relation:

$$\mathsf{R}_{\mathsf{eval}} = \left\{ \big(\mathbb{pp}, (\mathbf{cm}, \mathbf{x}, y), (f, st)\big) \in \mathbb{P} \times \mathbb{X} \times \mathbb{W} \;\middle|\; \begin{array}{c} (\mathbf{cm}, st) = \mathsf{Commit}(\mathbb{pp}, f) \\ \wedge \\ f(\mathbf{x}) = y \end{array} \right\}.$$

Because we operate over lattices, the knowledge soundness property allows the extractor to recover a polynomial $\widetilde{f}$, a state $\widetilde{st}$, and a bounded slack factor $\mathsf{c} \in \mathsf{SL}$. These extracted values must satisfy the evaluation relation up to the slack factor, i.e., $\mathsf{Open}(\mathbb{pp}, \mathbf{cm}, \widetilde{f}, \widetilde{st}, \mathsf{c}) = 1$ and the evaluation holds as $\widetilde{f}(\mathbf{x}) = y \cdot \mathsf{c}$.

2.6 Johnson-Lindenstrauss Projections Modulo q

Johnson-Lindenstrauss projections are (families of) linear maps $\Pi : \mathbb{R}^n \to \mathbb{R}^m$ such that for a fixed vector $\mathbf{x} \in \mathbb{R}^n$ the l_2 norm of $\Pi\mathbf{x}$ will be concentrated around that of $\mathbf{x}$ with overwhelming probability. They can also be shown to maintain this property when evaluated over prime fields. We recall the following useful heuristic result from [BS23].

Lemma 2.5 (Modular Johnson-Lindenstrauss). *Given a vector $\boldsymbol{x} \in [-q/2, q/2]^n$ with $\|\boldsymbol{x}\|_2 = b \leq q/125$ and $\Pi \in \{-1, 0, 1\}^{256\times n}$ where each entry is sampled i.i.d. from the distribution $\mathcal{D}$ such that*

$$\Pr[\mathcal{D} = 1] = \Pr[\mathcal{D} = -1] = 1/4 \qquad \textit{and} \qquad \Pr[\mathcal{D} = 0] = 1/2,$$

then it holds that

$$\Pr[\|\Pi\boldsymbol{x} \mod q\|_2 \leq \sqrt{30}b] \leq 2^{-128}$$
$$\Pr[\|\Pi\boldsymbol{x} \mod q\|_2 \geq \sqrt{337}b] \leq 2^{-128}$$

3 Grand Danois: Our Multilinear PCS

In this section, we present our main construction: a lattice-based succinct multilinear PCS. We formalize the protocol and present its components in a sequential manner.

Section organization. In section 3.1, we formally describe the Setup and Commit algorithms, detailing the generation of (tensor-structured) CRS matrices and the commitments. In section 3.2 we describe the interactive Eval protocol, which constitutes the bulk of our construction. We frame our evaluation protocol as a public coin interactive proof system, which can be made non-interactive using the Fiat-Shamir transform. Finally, in section 3.3 we provide concrete proof size estimates demonstrating the high efficiency of our scheme.

3.1 CRS Generation and Commitment Phase

We start by describing the CRS generation and commitment algorithms of our PCS for ℓ-variate multilinear polynomials.

Setup$(1^\lambda, \ell)$**.** Let $q = q(\lambda, \ell)$, $d = d(\lambda)$ and $n_A = n_B = n_D = O(1)$. Let $k = \lambda/\log q$, such that $\mathbb{F}_{q^k}$ is of exponential size, and let $\delta = \lceil \log_b q \rceil$ denote the base-b decomposition depth. Let $\ell = \ell_1 + \ell_2$, and set $L_i = 2^{\ell_i}$ for $i \in \{1, 2\}$ and $L = 2^\ell = L_1 L_2$. Set $r := L_2/d$ and let $\jmath$ be the value with $\jmath^2 = r$. Let $\mathcal{X}_{n,\mu}$ be the vSIS-friendly tensor distribution over $\mathcal{R}_q^{n\times 2^{\otimes\mu}}$ as defined in definition 2.2. The setup algorithm returns the public parameters $\mathbb{pp} = (\mathbf{A}, \mathbf{B}, \mathbf{D})$ sampled as follows:

- $\mathbf{A} \leftarrow\!\!\$\, \mathcal{X}_{n_A,\mu_A}$, where $2^{\mu_A} \geq \delta L_1$,

- $\mathbf{B} \leftarrow\!\$\ \mathcal{X}_{n_B,\mu_B}$, where $2^{\mu_B} \geq n_A \delta r$,
- $\mathbf{D} = [\mathbf{D}_1 \mid \mathbf{D}_2] \leftarrow\!\$\ \mathcal{X}_{n_D,\mu_D}$, where $2^{\mu_D} \geq \hat{\delta} \cdot 256r + \delta r$ (with $\mathbf{D}_1$ covering $\hat{\mathbf{p}}$ and $\mathbf{D}_2$ covering $\hat{\mathbf{w}}$).

$\mathsf{Commit}(\mathrm{pp}, f)$**.** A commitment to $f \in \mathbb{F}_q^{\leq 1}[X_1, \ldots, X_\ell]$ is computed as follows:

- Let $\mathbf{s} \in \mathcal{R}_q^{L_1 \times r}$ denote the matrix obtained by packing the 2^ℓ coefficients of f into blocks of size d. We define the columns as $\mathbf{s}_i \in \mathcal{R}_q^{L_1}$ for $i \in [r]$.
- $\hat{\mathbf{s}}_i := \mathbf{G}_{L_1}^{-1}(\mathbf{s}_i) \in \mathcal{R}_q^{\delta L_1}$ for $i \in [r]$.
- $\mathbf{t}_i := \mathbf{A}\hat{\mathbf{s}}_i \in \mathcal{R}_q^{n_A}$ for $i \in [r]$.
- $\hat{\mathbf{t}}_i := \mathbf{G}_{n_A}^{-1}(\mathbf{t}_i)$ for $i \in [r]$.
- Return the commitment $\mathbf{cm}$ defined as

$$\mathbf{cm} := \mathbf{B} \begin{bmatrix} \hat{\mathbf{t}}_1 \\ \vdots \\ \hat{\mathbf{t}}_r \end{bmatrix} \in \mathcal{R}_q^{n_B},$$

with decommitment state $st := (\hat{\mathbf{s}}_i, \hat{\mathbf{t}}_i)_{i \in [r]}$.

3.2 Polynomial Evaluation Protocol $\mathsf{Eval} = \langle \mathcal{P}_{\mathsf{eval}}, \mathcal{V}_{\mathsf{eval}} \rangle$

The prover $\mathcal{P}_{\mathsf{eval}}$ aims to convince the verifier $\mathcal{V}_{\mathsf{eval}}$ of the evaluation claim $y = f(x_1, \ldots, x_\ell) \in \mathbb{F}_{q^k}$. We consider the vector of evaluations of f over the Boolean hypercube. This vector can naturally be embedded into a factor d shorter vector over $\mathcal{R}_q$ in the coefficients of $\mathcal{R}_q$ elements. We denote by $\mathbf{s}$ this embedded vector. This evaluation can be expressed as

$$y = f(x_1, \ldots, x_\ell) = \mathbf{a}^\top \left[f(0 \ldots 0) \mid \cdots \mid f(1 \ldots 1) \right] \mathbf{b}.$$

We define $\mathbf{w}$ to be $\mathbf{a}^\top [\mathbf{s}_1 | \ldots | \mathbf{s}_r] \in \mathcal{R}_{q^k}^r$ and $\hat{\mathbf{w}} = \mathbf{G}_r^{-1}\mathbf{w}$. From this point, we follow the same approach as in Greyhound, reducing validation of the evaluation claim to proving knowledge of short vectors $(\hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$ that satisfy the following system of linear constraints over $\mathcal{R}_q$:

$$\begin{bmatrix} \mathbf{D} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{B} & \mathbf{0} \\ \mathbf{I}_k \otimes \mathbf{c}^\top \mathbf{G}_r & \mathbf{0} & -\mathbf{a}^\top \mathbf{G}_{L_1} \hat{\mathbf{G}}_{\delta L_1} \\ \mathbf{0} & \mathbf{c}^\top \otimes \mathbf{G}_{n_A} & -\mathbf{A}\hat{\mathbf{G}}_{\delta L_1} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{w}} \\ \hat{\mathbf{t}} \\ \hat{\mathbf{z}} \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ \mathbf{cm} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

and the following constraint over $\mathbb{F}_{q^k}$:

$$\langle \mathbf{b}^\top \mathbf{G}_r, \mathsf{cf}(\hat{\mathbf{w}}) \rangle = y.$$

In the constraint system, we consider $\mathbf{a}^\top$ as a k-row matrix. Firstly, we want to prove the shortness of $(\hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$ using random projections, so our first step is to integrate such a check into the above constraints. To make sure this is compatible with succinct verification, we impose structure on the random projection. We will do this in line with the approach given in RoK and Roll [KLNO25] albeit taking it a step further. To this end, we make use of a result from recent work [KLN+26].

Lemma 3.1. *Let $\boldsymbol{w} \in \mathbb{Z}^n$ with $\|\boldsymbol{w}\|_2 \leq q/125$ and $q \in \mathbb{N}$ and let $\Pi \in \{-1,0,1\}^{256\times m}$ be sampled as in lemma 2.5 and define $\Pi' = \boldsymbol{I}_{n/m} \otimes \Pi$. Then, it holds that*

$$\Pr\left[\frac{\|\Pi'\boldsymbol{w} \bmod q\|_2}{\|\boldsymbol{w}\|_2} \notin [\sqrt{30}, \sqrt{337}]\right] \leq 2^{-128} n/m \tag{12}$$

From lemma 3.1 we derive the following lemma.

Lemma 3.2. *Let $\boldsymbol{w} \in \mathbb{Z}^n$ with $\|\boldsymbol{w}\|_2 \leq \frac{q}{125\sqrt{337}}$ and $q \in \mathbb{N}$ and let $\Pi, \Pi' \in \{-1,0,1\}^{256\times m}$ be sampled as in lemma 2.5. Define*

$$\hat{\Pi} = \boldsymbol{I}_{n/m} \otimes \Pi \quad \textit{and} \quad \hat{\Pi}' = \boldsymbol{I}_{n/m^2} \otimes \Pi'.$$

Then, it holds that

$$\Pr\left[\frac{\|\hat{\Pi}'\hat{\Pi}\boldsymbol{w} \bmod q\|_2}{\|\boldsymbol{w}\|_2} \notin [30, 337]\right] \leq 2^{-128}\frac{nm+n}{m^2}. \tag{13}$$

Proof. The proof is a union bound on the probabilities of the events that $\frac{\|\hat{\Pi}\mathbf{w} \bmod q\|_2}{\|\mathbf{w}\|_2} \notin [\sqrt{30}, \sqrt{337}]$ and $\frac{\|\hat{\Pi}'\hat{\Pi}\mathbf{w} \bmod q\|_2}{\|\hat{\Pi}\mathbf{w} \bmod q\|_2} \notin [\sqrt{30}, \sqrt{337}]$, since at least one of them must occur for it to be the case that $\frac{\|\mathbf{w}\|_2}{\|\hat{\Pi}'\hat{\Pi}\mathbf{w} \bmod q\|_2} \notin [30, 337]$. From the bound in eq. (12) we get

$$2^{-128}\frac{n}{m} + 2^{-128}\frac{n}{m^2} = 2^{-128}(\frac{nm+n}{m^2}).$$

□

We will use lemma 3.2 to sample projections that have succinct representations but allow for more aggressive dimensionality reduction of the witness. Once the verifier has sampled projection matrices $\mathbf{J}, \mathbf{J}' \in \{-1,0,1\}^{256\times 256_J}$, the prover needs to apply $(\mathbf{I}_r \otimes \mathbf{J}')(\mathbf{I}_{rJ} \otimes \mathbf{J}) = \Pi$ to $\mathbf{s}$. Here, the entries of $\mathbf{J}, \mathbf{J}'$ are implicitly embedded into $\mathcal{R}_q$ as constant polynomials. Let us call the result $\mathbf{p} = \Pi\mathbf{s}$. We note that $\mathbf{p}$ can be computed as $\hat{\mathbf{J}}[\mathbf{s}_1|\ldots|\mathbf{s}_r]$ for a smaller $\hat{\mathbf{J}} := \mathbf{J}'(\mathbf{I}_J \otimes \mathbf{J})$. Thus, it has a structure that fits into our reference point for proving evaluations of tensor-structured linear forms on f. The short witness now becomes $(\hat{\mathbf{p}}, \hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$ and it should satisfy the following linear constraints:

$$\begin{bmatrix} \mathbf{D}_1 & \mathbf{D}_2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{B} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_k \otimes (\mathbf{c}^\top\mathbf{G}_r) & \mathbf{0} & -\mathbf{a}^\top\mathbf{G}_{L_1}\hat{\mathbf{G}}_{\delta L_1} \\ (\mathbf{c}^\top\mathbf{G}_{L_1}) & \mathbf{0} & \mathbf{0} & -\hat{\mathbf{J}}\hat{\mathbf{G}}_{\delta L_1} \\ \mathbf{0} & \mathbf{0} & \mathbf{c}^\top \otimes \mathbf{G}_{n_A} & -\mathbf{A}\hat{\mathbf{G}}_{\delta L_1} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{p}} \\ \hat{\mathbf{w}} \\ \hat{\mathbf{t}} \\ \hat{\mathbf{z}} \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ \mathbf{cm} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \tag{14}$$

Furthermore, it should hold that

$$\langle \mathbf{b}^\top\mathbf{G}_r, \mathsf{cf}(\hat{\mathbf{w}})\rangle = y \tag{15}$$

over $\mathbb{F}_{q^k}$. We now want to prove knowledge of such a $(\hat{\mathbf{p}}, \hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$.

3.2.1 Proving Matrix-Vector Products over $\mathcal{R}_q$ using Sumcheck. To prove knowledge of $(\hat{\mathbf{p}}, \hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$ we employ a similar approach as that of Hachi [NOZ26]. There, the authors consider the general case of an unstructured $n \times m$ matrix $\mathbf{M}$ over $\mathcal{R}_q$ and the claim that $\mathbf{Mx} = \mathbf{y}$ for a small vector $\mathbf{x}$ over $\mathcal{R}_q$. The main observation is that for $\mathbf{Mx} = \mathbf{y}$ to hold over $\mathcal{R}_q$ it must be the case that there exists $\mathbf{r} \in (\mathbb{F}_q[X]^{<d})^n$ such that

$$\mathbf{Mx} = \mathbf{y} + (X^d + 1)\mathbf{r}, \tag{16}$$

where the pairwise multiplications are taken over $\mathbb{F}_q[X]$ rather than $\mathcal{R}_q$. This equality, allows the authors construct a sumcheck protocol over the multilinear extensions of the coefficient representations of $\mathbf{M}, \mathbf{y}, (\mathbf{x}, \mathbf{r})$ such that the verifier accepts only if eq. (16) holds.

We also construct a sumcheck protocol over the coefficients of the $\mathcal{R}_q$ elements making up our witness. However, instead of considering multiplications of polynomials in $\mathbb{F}_q[X]$ we utilize the matrix representation of multiplication by fixed elements in $\mathcal{R}_q$. Using the function

$$\begin{aligned} \mathsf{rot} : \mathbb{F}_q^d &\to \mathbb{F}_q^d \\ \mathsf{rot}(a_0, \dots, a_{d-1}) &= (-a_{d-1}, a_0, \dots, a_{d-2}) \end{aligned}$$

multiplication by an element $a \in \mathcal{R}_q$ can be expressed by the matrix

$$\mathbf{M}_a := [\mathsf{cf}(a), \mathsf{rot}(\mathsf{cf}(a)), \mathsf{rot}^2(\mathsf{cf}(a)), \dots, \mathsf{rot}^{d-1}(\mathsf{cf}(a))] \in \mathbb{F}_q^{d \times d}. \tag{17}$$

Proving the correctness of matrix-vector products can be done via a common approach that is neatly compatible with the sumcheck-protocol. The verifier samples a random element $\alpha \in \mathbb{F}_{q^k}$ defining the vector $\boldsymbol{\alpha} = (1, \alpha, \alpha^2, \dots, \alpha^{d-1})$. Then the verifier needs to be convinced that

$$\boldsymbol{\alpha}\mathbf{Mx} = \boldsymbol{\alpha}\mathbf{y}$$

and, folding $\boldsymbol{\alpha}$ into the matrix, $\boldsymbol{\alpha}\mathbf{Mx}$ can be expressed as the sum

$$\sum_{b \in \{0,1\}^{\log d}} \widetilde{(\boldsymbol{\alpha}\mathbf{M})}(b)\, \tilde{\mathbf{x}}(b).$$

Furthermore, given the structure of $\mathbf{M}_a$ the vector $\boldsymbol{\alpha}\mathbf{M}$ can be computed in $O(d)$ time meaning that the sum only needs to range over $\{0,1\}^{\log d}$. Our approach is now to utilize this formulation of multiplication over $\mathcal{R}_q$ along with the structure of the public parameters of a vector commitment scheme based on the vSIS assumption to obtain a sumcheck-based protocol for the claim $\mathbf{Mx} = \mathbf{y}$ where $\mathbf{M}$ is a vSIS matrix. We can define

$${}_{\beta}\mathbf{M}_{\alpha} := \boldsymbol{\beta} \begin{bmatrix} \boldsymbol{\alpha}\mathbf{M}_{1,1} & \boldsymbol{\alpha}\mathbf{M}_{1,2} & \dots & \boldsymbol{\alpha}\mathbf{M}_{1,m} \\ & \vdots & & \\ \boldsymbol{\alpha}\mathbf{M}_{n,1} & \boldsymbol{\alpha}\mathbf{M}_{n,2} & \dots & \boldsymbol{\alpha}\mathbf{M}_{n,m} \end{bmatrix} \tag{18}$$

where each $\mathbf{M}_{i,j}$ is the multiplication matrix corresponding to entry (i, j) of $\mathbf{M}$ and $\boldsymbol{\beta} := (1, \beta, \beta^2, \ldots, \beta^{n-1})$ for a random $\beta \in \mathbb{F}_{q^k}$ sampled by the verifier. Then, we want to verify that ${}_{\beta}\mathbf{M}_{\alpha}\mathbf{x} = \boldsymbol{\beta}\mathbf{y}$. To evaluate the left-hand side we will employ the sumcheck protocol. Thus, the verifier must evaluate $\widetilde{{}_{\beta}\mathbf{M}_{\alpha}}$ in a random point. For the point $\mathbf{e} \in \mathbb{F}_{q^k}^{\log md}$ this evaluation can be expressed as

$$(\boldsymbol{\beta} \otimes \boldsymbol{\alpha}) \begin{bmatrix} \prod_{i=1}^{\log m} (\mathbf{I}_d(1 - e_i) + \mathbf{M}_{1,i} e_i) \\ \vdots \\ \prod_{i=1}^{\log m} (\mathbf{I}_d(1 - e_i) + \mathbf{M}_{n,i} e_i) \end{bmatrix} \mathbf{e}'$$

where $\mathbf{e}'$ is the evaluation vector for the remaining $\log md - \log m$ entries in $\mathbf{e}$. We now note that each block in the matrix given in the above expression can be computed via $\log m$ multiplications over $\mathcal{R}_{q^k}$. Now, the final evaluation can be computed as before.

It remains to verify that $\langle \mathbf{b}^\top \mathbf{G}_r, \mathsf{cf}(\hat{\mathbf{w}}) \rangle = y$ over $\mathbb{F}_{q^k}$. In Hachi [NOZ26] the authors develop a method by which to embed $\mathbb{F}_{q^k}$ elements into $\mathcal{R}_q$ elements in a manner such that relations over $\mathbb{F}_{q^k}$ can be expressed over $\mathcal{R}_q$. Due to the communication cost of this approach we opt for a different approach. This choice means that we can only support commitments to polynomials with coefficients over $\mathbb{F}_q$ rather than $\mathbb{F}_{q^k}$ unless we employ additional techniques. Since we only support commitments to polynomials over $\mathbb{F}_q$ we only have access to the individual $\mathbb{F}_q$ coefficients of each entry in $\hat{\mathbf{w}}$. However, there does exist a tensor-structured linear form $\boldsymbol{\sigma}$ over $\mathbb{F}_{q^k}$ that maps the $\mathbb{F}_q$ coefficients of $\hat{\mathbf{w}}$ to $\langle \mathbf{b}^\top \mathbf{G}_r, \mathsf{cf}(\hat{\mathbf{w}}) \rangle$, which we can employ in a straightforward fashion in an invocation of sumcheck by 0-padding $\boldsymbol{\sigma}$ to the length of the full witness. Since $\boldsymbol{\sigma}$ is both tensor-structured and has length proportional to the splitting factor of the current round the verifier can easily evaluate its multilinear extension by itself. Furthermore, these two invocations of sumcheck can be batched via a random verifier-sampled challenge producing the sum

$$\sum_i \widetilde{{}_{\beta}\mathbf{M}_{\alpha}}(i)\widetilde{\mathbf{z}'}(i) + \gamma \sum_i \tilde{\boldsymbol{\sigma}}(i)\widetilde{\mathbf{z}'}(i) = H, \tag{19}$$

where $\mathbf{z}' = (\hat{\mathbf{p}}, \hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$, and $H := (\boldsymbol{\beta} \otimes \boldsymbol{\alpha})\mathsf{cf}(\mathbf{y}) + \gamma y$ for $\mathbf{y} = [\mathbf{v}, \mathbf{cm}, \mathbf{0}, \mathbf{0}, \mathbf{0}]^\top$ the right-hand side of eq. (14). Of course, in our setting the matrix $\mathbf{M}$ is not just a vSIS matrix but rather the one defined in eq. (14). Thus, we also need to ensure that the rows corresponding to $\hat{\mathbf{J}}$ and $\mathbf{a}$ fit into our strategy.

3.2.2 Verifier-Efficient Evaluation of $\widetilde{{}_{\boldsymbol{\beta}}\mathbf{M}_{\boldsymbol{\alpha}}}$. Consider the rows $\hat{\mathbf{J}}$. $\hat{\mathbf{J}}$ has the structure that it is a diagonal block matrix with the block being $\mathbf{J}'(I_\jmath \otimes \mathbf{J})$. Thus, we focus on $\mathbf{J}'(I_\jmath \otimes \mathbf{J})$. Technically, we are implicitly embedding the entries of $\mathbf{J}'(\mathbf{I}_\jmath \otimes \mathbf{J})$ into $\mathcal{R}_q$ as constant polynomials and wish to compute a partial evaluation in line with the rest of $\mathbf{M}$. $\mathbf{J}'(I_\jmath \otimes \mathbf{J})$ is a $256 \times 256\jmath^2$ matrix. Each partial evaluation is the inner product between a row of $\mathbf{J}'(I_\jmath \otimes \mathbf{J})$ and the

evaluation vector corresponding to the evaluation point $\mathbf{e}$. We can split $\mathbf{e}$ into $\mathbf{e}' \otimes \mathbf{e}''$ where $\mathbf{e}'$ matches the width of $\mathbf{J}$. Now, we can compute $\mathbf{J}\mathbf{e}'$. The vector of partial evaluations is then $\mathbf{J}'((\mathbf{J}'e') \otimes e'')$. Each of these can now be considered as a constant polynomial in $\mathcal{R}_{q^k}$ whose multiplication matrix can be handled very easily. Lastly, we have $\mathbf{a}^\top$. In our constraint system, $\mathbf{a}$ is a matrix over $\mathbb{F}_q$ with k rows, the entries of which we implicitly regard as constant polynomials in $\mathcal{R}_q$. However, this formulation removes the tensor-structure exhibited by $\mathbf{a}$ over $\mathbb{F}_{q^k}$. Our solution is simply to employ the same technique as for the full $\mathcal{R}_q$ rows except over $\mathbb{F}_{q^k}$. Multiplying by an element $a \in \mathbb{F}_{q^k}$ can be expressed by a similar $k \times k$ matrix. This domain maintains our desired tensor-structure. Furthermore, for an evaluation point $\mathbf{e}$ after computing

$$((1 - \mathbf{e}_1) + \mathbf{e}_1\mathbf{a}_1)((1 - \mathbf{e}_2) + \mathbf{e}_2\mathbf{a}_2) \dots$$

the first column of the resulting matrix contains the k evaluations that we are interested in. Once again, these can be viewed as constant polynomials in $\mathcal{R}_{q^k}$ and be handled appropriately.

3.2.3 Putting Things Together. We now present the full protocol. We wish to construct a protocol for the following relation:

$$\mathsf{R} = \left\{ \begin{array}{l} \big((\mathbf{A}, \mathbf{B}, \mathbf{D}), (\mathbf{cm}, \mathbf{x}, y), (f, \mathbf{t})\big) \in \mathbb{P} \times \mathbb{X} \times \mathbb{W} \\ \\ \mathbb{P} := \mathcal{R}_q^{n_A \times \delta L_1} \times \mathcal{R}_q^{n_B \times n_A \delta r} \times \mathcal{R}_q^{n_D \times (\hat{\delta} 256 r + \delta r)}, \\ \mathbb{X} := \mathcal{R}_q^{n_B} \times \mathbb{F}_{q^k}^{\ell} \times \mathbb{F}_{q^k}, \\ \mathbb{W} := \mathbb{F}_q^{L} \times \mathcal{R}_q^{n_A r} \\ \\ \end{array} \middle| \begin{array}{c} \mathbf{s} = \mathsf{pack}(f) \\ \hat{\mathbf{s}}_i = \mathbf{G}_{L_1}^{-1}(\mathbf{s}_i) \\ \mathbf{t}_i = \mathbf{A}\hat{\mathbf{s}}_i \\ f(\mathbf{x}) = y \\ \|\hat{\mathbf{s}}_i\|_2 \leq \omega \\ \mathbf{cm} = \mathbf{B}\hat{\mathbf{t}} \end{array} \right\}.$$

It is straightforward to see that R can be reduced to a relation capturing polynomial commitments. We present the protocol assuming access to a norm-bounded polynomial commitment scheme. Under this abstraction, the protocol can be recursively composed with itself. This also gives the flexibility to compose with alternative schemes, for example LaBRADOR [BS23] at a certain point as is often done in modern lattice-based polynomial commitment schemes. The protocol Π_{eval} is given in fig. 1.

Theorem 3.3. *Π_{eval} is complete and knowledge sound for the relation R with norm slack $\sqrt{337/30} \approx 3.35$.*

Proof. Completeness follows directly, so we focus on knowledge soundness. Our starting point is the analysis given in [NS24] where they show that the protocol that serves as our starting point satisfies the relaxed notion of knowledge soundness, wherein the knowledge extractor can extract weak openings to each $\mathbf{t}_i$ that satisfy the given evaluation claim. This follows from the binding property of Ajtai commitments under the SIS assumption and from CWSS. Namely,

Public Parameters: $\mathbf{A} \leftarrow\$ \mathcal{X}_{n_A,\mu_A}$, $\mathbf{B} \leftarrow\$ \mathcal{X}_{n_B,\mu_B}$, $\mathbf{D} = [\mathbf{D}_1 \mid \mathbf{D}_2] \leftarrow\$ \mathcal{X}_{n_D,\mu_D}$
Prover's Input: $f \in \mathbb{F}_q^{\leq 1}[X_1, \ldots, X_\ell]$, state $st = (\hat{\mathbf{s}}_i, \hat{\mathbf{t}}_i)_{i\in[r]}$ from Commit
Verifier's Input: $\mathbf{cm} \in \mathcal{R}_q^{n_B}$, $y \in \mathbb{F}_{q^k}$, $\mathbf{a} \in \mathbb{F}_{q^k}^{L_1}, \mathbf{b} \in \mathbb{F}_{q^k}^{r}$

$\mathcal{P}$		$\mathcal{V}$
	$\xleftarrow{\mathbf{J},\mathbf{J}'}$	$\mathbf{J}, \mathbf{J}' \leftarrow \mathcal{D}$ (lemma 2.5)
$\mathbf{p} = \Pi\mathbf{s}$		
$\mathbf{w} = \mathbf{a}^\top \left[\mathbf{s}_1 \mid \cdots \mid \mathbf{s}_r\right]$		
$\hat{\mathbf{p}} = \hat{\mathbf{G}}_{L_1}^{-1}(\mathbf{p})$		
$\hat{\mathbf{w}} = \hat{\mathbf{G}}_{r}^{-1}(\mathbf{w})$		
$\mathbf{v} = \mathbf{D}(\hat{\mathbf{p}}, \hat{\mathbf{w}})^\top$	$\xrightarrow{\mathbf{v}}$	
	$\xleftarrow{\mathbf{c}}$	$\mathbf{c} \leftarrow \mathcal{C}^r$
$\mathbf{z} = \sum_{i=1}^{r} c_i \mathbf{s}_i$		
$\hat{\mathbf{z}} = \hat{\mathbf{G}}_{\delta L_1}^{-1}(\mathbf{z})$		
$\mathbf{z}' = (\hat{\mathbf{p}}, \hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$		
$\mathbf{cm}' = \mathsf{Commit}(\mathbf{z}')$	$\xrightarrow{\mathbf{cm}'}$	
	$\xleftarrow{\alpha,\beta,\gamma}$	$\alpha, \beta, \gamma \leftarrow \mathbb{F}_{q^k}$
run sumcheck protocol for eq. (19): $\sum_i \widetilde{{}_\beta \mathbf{M}_\alpha}(i)\widetilde{\mathbf{z}'}(i) + \gamma \sum_i \tilde{\boldsymbol{\sigma}}(i)\widetilde{\mathbf{z}'}(i) = H$, where $\widetilde{{}_\beta \mathbf{M}_\alpha}, \tilde{\boldsymbol{\sigma}}$ are defined as in section 3.2.1 and $H = (\boldsymbol{\beta} \otimes \boldsymbol{\alpha})\mathsf{cf}(\mathbf{y}) + \gamma y$. run PoK of opening $\mathbf{z}' = (\hat{\mathbf{p}}, \hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$ that satisfies the evaluation claim from sumcheck and norm bounds $\sqrt{337/30}\omega$ for $\mathbf{p}_i$ and $\sqrt{b^2(n_A L_2 \delta d + L_2 \delta d) + (L_2 \kappa b')^2 L_1 d}$ for $(\hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$		**Accept** if the sumcheck accepts, and PoK accepts

Fig. 1: Protocol Π_{eval} for relation $\mathsf{R}_{\mathsf{eval}}$

we can argue that any two accepting transcripts must contain the same $\hat{\mathbf{t}}$ since otherwise they directly constitute a break of the commitment $\mathbf{cm}$. From there, after recomposing appropriately we can use CWSS to extract weak openings to each $\mathbf{t}_i$. Using this result and via two applications of the Schwartz-Zippel lemma we can argue that for an alternative protocol where the prover simply sends $(\hat{\mathbf{p}}, \hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$ after receiving $\mathbf{c}$ in plaintext and the verifier checks eq. (14) directly and checks that $\|\mathbf{p}_i\|_2 \leq 337\omega$ there exists a knowledge extractor that can extract weak openings $\mathbf{s}_i$ to each $\mathbf{t}_i$ that satisfy the claimed polynomial evaluation and where the recombination of $\hat{\mathbf{p}}$ is the projection using Π derived from $\mathbf{J}, \mathbf{J}'$. By having the verifier check that $\|\mathbf{p}_i\|_2 \leq 337\omega$ we can use lemma 3.2 to argue that the l_2 norm of each of the extracted weak openings is less than $\sqrt{337/30}\omega$ with probability at least

$$1 - d2^{-128}\frac{Lj^2 + L}{j^4},$$

where the factor d accounts for a union bound over the d coefficients of each ring element. Now, by the fact that compositions of compatible proofs of knowledge maintain knowledge soundness [BS23] we maintain knowledge soundness by instead proving knowledge of $(\hat{\mathbf{p}}, \hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$ that satisfy eq. (14) and a norm bound for each limb derived from that on $\hat{\mathbf{p}}$ and the rest of $\hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}}$. Depending on how the limbs are packed during recursion this may increase the slack beyond the amount incurred by lemma 3.2, since the limbs containing the first and last projections may also contain parts of $\hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}}$. However, it is always possible to ensure that this does not happen with appropriate padding/alignment. In particular, $|\hat{\mathbf{p}}|$ will always be a power of 2 and $|\hat{\mathbf{p}}|$ can always be placed at the beginning of the witness vector. Proving evaluation consistency is done probabilistically using the sumcheck protocol run over $\mathbb{F}_{q^k}$ and knowledge soundness then follows from the knowledge soundness of the invoked proof of knowledge and the soundness of the sumcheck protocol. □

We further note that if we are willing to accept that the knowledge extractor is only able to extract weak openings without a norm guarantee from a Johnson-Lindenstrauss projection for the top layer (as is the case, e.g., in Greyhound) then we can omit the application of the random projection for the first invocation of Π_{eval}.

3.2.4 Compressing the Witness when $L_1 \leq 256$. As in RoK and Roll, since we embed embed our random projection into $\mathcal{R}_q$ we need L_1 to be larger than 256 to be able to produce projections smaller than the current witness size. For a given value of r, once we've compressed a number of times it will eventually be the case that L_1 approaches or drops below 256. In this case we can employ two strategies. Firstly, when recursing we can switch rings moving to a cyclotomic ring with smaller dimension d and thus increasing L_1. Secondly, once this is no longer feasible but while our witness size is still $O(\lambda^2)$ $\mathbb{F}_q$ elements we can use our method for sampling a random projection matrix to produce a projection for the entire witness. The matrix-vector product between $(\hat{\mathbf{p}}, \hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$ and this

projection matrix can be verified via an invocation of sumcheck, which we can batch with the other instances, and using the techniques from section 3.2.2 the multilinear extension of this projection matrix can be evaluated in $O(\lambda^2)$ time. Via a norm bound on the projection, we can argue that the witness must be small and using Greyhound as the underlying PCS, an evaluation can be verified in $O(\lambda^2)$ time.

3.3 Proof Sizes

In this section we provide estimates of proof sizes of our protocol based on parameters used in previous work. We leave it to future work to provide an implementation and finding optimal choices of concrete parameters. We assume throughout that we have an instantiation of Π_{eval} that is rendered non-interactive using the Fiat-Shamir heuristic and thus we do not count any challenges sent by the verifier. For a 32-bit prime q, $k = 4$, and using the estimate that an Ajtai commitment is 4KB we have for a single invocation of Π_{eval} 8KB of commitment material from $\mathbf{v}$ and $\mathbf{cm}$. Then we have an invocation of the sumcheck protocol over a polynomial of individual degree at most 2. Assuming that the length of $\mathbf{z}' = (\hat{\mathbf{p}}, \hat{\mathbf{w}}, \hat{\mathbf{t}}, \hat{\mathbf{z}})$ is at most 2^{32} $\mathbb{F}_q$ elements, this yields $4 \cdot 32 \cdot 3 \cdot 32\mathsf{bits} = 1.5\mathsf{KB}$ of communication. Assuming for instance that we have a splitting factor and decomposition parameters such that $\mathbf{z}'$ is at least 32 or 64 times smaller in length than f, we only need around 5-6 or less invocations to bring a witness of size 2^{32} down to a size where composing with LaBRADOR or Greyhound becomes more efficient. Based on these estimates we conclude that even for witnesses of size 2^{32} $\mathbb{F}_q$ elements our proof sizes should sit between 80 and 90 KB. This is considerably smaller than prior work achieving polylogarithmic verification complexity. We note that Hachi [NOZ26] reports smaller proofs of approximately 55KB, but at square-root rather than polylogarithmic verification complexity; our construction trades a modest increase in proof size for the asymptotically faster verifier. We also note that our estimate is rather conservative and for appropriately chosen parameters sizes could potentially be even smaller.

Acknowledgment. We thank Quang Dao for pointing out a soundness issue in the ring-switching step of an earlier version of this work, which this second version addresses. The authors were supported by the European Research Council (ERC) under the EU's Horizon 2020 research and innovation programme (Grant Agreement No. 101124977, DECRYPSYS). The second author was additionally supported by the Danish Innovation Fund (Grant No. 4298-00014A).

References

BBB+18. Benedikt Bünz, Jonathan Bootle, Dan Boneh, Andrew Poelstra, Pieter Wuille, and Greg Maxwell. Bulletproofs: Short proofs for confidential transactions and more. In *2018 IEEE Symposium on Security and Privacy*, pages 315–334. IEEE Computer Society Press, May 2018.

BBHR18. Eli Ben-Sasson, Iddo Bentov, Yinon Horesh, and Michael Riabzev. Fast reed-solomon interactive oracle proofs of proximity. In Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella, editors, *ICALP 2018*, volume 107 of *LIPIcs*, pages 14:1–14:17. Schloss Dagstuhl, July 2018.

BC25a. Dan Boneh and Binyi Chen. LatticeFold: A lattice-based folding scheme and its applications to succinct proof systems. In Goichiro Hanaoka and Bo-Yin Yang, editors, *ASIACRYPT 2025, Part III*, volume 16247 of *LNCS*, pages 330–362. Springer, Singapore, December 2025.

BC25b. Dan Boneh and Binyi Chen. LatticeFold+: Faster, simpler, shorter lattice-based folding for succinct proof systems. In Yael Tauman Kalai and Seny F. Kamara, editors, *CRYPTO 2025, Part VII*, volume 16006 of *LNCS*, pages 327–361. Springer, Cham, August 2025.

BCCT13. Nir Bitansky, Ran Canetti, Alessandro Chiesa, and Eran Tromer. Recursive composition and bootstrapping for SNARKS and proof-carrying data. In Dan Boneh, Tim Roughgarden, and Joan Feigenbaum, editors, *45th ACM STOC*, pages 111–120. ACM Press, June 2013.

BS23. Ward Beullens and Gregor Seiler. LaBRADOR: Compact proofs for R1CS from module-SIS. In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part V*, volume 14085 of *LNCS*, pages 518–548. Springer, Cham, August 2023.

CLM23. Valerio Cini, Russell W. F. Lai, and Giulio Malavolta. Lattice-based succinct arguments from vanishing polynomials - (extended abstract). In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part II*, volume 14082 of *LNCS*, pages 72–105. Springer, Cham, August 2023.

CT10. Alessandro Chiesa and Eran Tromer. Proof-carrying data and hearsay arguments from signature cards. In Andrew Chi-Chih Yao, editor, *ICS 2010*, pages 310–331. Tsinghua University Press, January 2010.

FMN23. Giacomo Fenzi, Hossein Moghaddas, and Ngoc Khanh Nguyen. Lattice-based polynomial commitments: Towards asymptotic and concrete efficiency. Cryptology ePrint Archive, Report 2023/846, 2023.

GLS+23. Alexander Golovnev, Jonathan Lee, Srinath T. V. Setty, Justin Thaler, and Riad S. Wahby. Brakedown: Linear-time and field-agnostic SNARKs for R1CS. In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part II*, volume 14082 of *LNCS*, pages 193–226. Springer, Cham, August 2023.

HMZ25. Mi-Ying Miryam Huang, Xinyu Mao, and Jiapeng Zhang. Sublinear proofs over polynomial rings. Cryptology ePrint Archive, Report 2025/199, 2025.

KLN+26. Michael Klooss, Russell WF Lai, Ngoc Khanh Nguyen, Michał Osadnik, and Lorenzo Tucci. Rokoko: Lattice-based succinct arguments, a committed refinement. *Cryptology ePrint Archive*, 2026.

KLNO25. Michael Klooß, Russell W. F. Lai, Ngoc Khanh Nguyen, and Michal Osadnik. RoK and roll - verifier-efficient random projection for $\tilde{O}(\lambda)$-size lattice arguments - (extended abstract). In Goichiro Hanaoka and Bo-Yin Yang, editors, *ASIACRYPT 2025, Part III*, volume 16247 of *LNCS*, pages 297–329. Springer, Singapore, December 2025.

KZG10. Aniket Kate, Gregory M. Zaverucha, and Ian Goldberg. Constant-size commitments to polynomials and their applications. In Masayuki Abe, editor, *ASIACRYPT 2010*, volume 6477 of *LNCS*, pages 177–194. Springer, Berlin, Heidelberg, December 2010.

Lee21. Jonathan Lee. Dory: Efficient, transparent arguments for generalised inner products and polynomial commitments. In Kobbi Nissim and Brent Waters, editors, *TCC 2021, Part II*, volume 13043 of *LNCS*, pages 1–34. Springer, Cham, November 2021.

NOZ26. Ngoc Khanh Nguyen, George O'Rourke, and Jiapeng Zhang. Hachi: Efficient lattice-based multilinear polynomial commitments over extension fields. *IACR Cryptol. ePrint Arch.*, 2026:156, 2026.

NS24. Ngoc Khanh Nguyen and Gregor Seiler. Greyhound: Fast polynomial commitments from lattices. In Leonid Reyzin and Douglas Stebila, editors, *CRYPTO 2024, Part X*, volume 14929 of *LNCS*, pages 243–275. Springer, Cham, August 2024.

NS25. Wilson Nguyen and Srinath Setty. Neo: Lattice-based folding scheme for CCS over small fields and pay-per-bit commitments. Cryptology ePrint Archive, Report 2025/294, 2025.

NS26. Wilson Nguyen and Srinath T. V. Setty. Neo and superneo: Post-quantum folding with pay-per-bit costs over small fields. *IACR Cryptol. ePrint Arch.*, 2026:242, 2026.

Val08. Paul Valiant. Incrementally verifiable computation or proofs of knowledge imply time/space efficiency. In Ran Canetti, editor, *TCC 2008*, volume 4948 of *LNCS*, pages 1–18. Springer, Berlin, Heidelberg, March 2008.

WTs+18. Riad S. Wahby, Ioanna Tzialla, abhi shelat, Justin Thaler, and Michael Walfish. Doubly-efficient zkSNARKs without trusted setup. In *2018 IEEE Symposium on Security and Privacy*, pages 926–943. IEEE Computer Society Press, May 2018.