

# CauchyFold: Residue-Optimal High-Arity Lattice Folding via Scaled Cauchy Challenges

Xiang Wang

Guizhou University, Guiyang, China  
ee.wangx24@gzu.edu.cn

**Abstract.** Folding schemes combine a batch of relation instances into one accumulator. For quadratic relations, applying the fold directly creates a mixed term for every pair of inputs. We present CauchyFold, a lattice protocol that folds one accumulator with  $k$  fresh quadratic instances without enumerating these pairwise terms.

The companion work [WX26] uses Cauchy folding coefficients to represent the mixed contribution through a vector-valued polynomial of degree below  $k$ . We give two algorithms that compute its coefficients with quasi-linear arithmetic in  $k$  for fixed relation dimensions. The protocol commits this polynomial before the folding challenge. Since the folding coefficients need not be short integers, it computes the fold over an extension field and canonically re-encodes the result as bits. This preserves the accumulator’s commitment layout and honest opening bound.

For classical resettable provers, we give explicit extraction-error and expected-time bounds for a folding node with a fixed-depth reduction chain. It recovers valid source openings or a short nonzero vector in the kernel of a commitment matrix. A compiled profile at  $k = 1024$  uses 162.67 KiB of no-retry interactive communication, excluding incoming commitments and the CRS.

**Keywords:** Folding schemes · Lattice-based cryptography · Cauchy interpolation · Knowledge extraction

## 1 Introduction

Recursive proof systems can use folding to absorb new computation into an existing accumulator [BCMS20, KST22]. Each fold combines several relation instances into one accumulated instance that can participate in the next fold. Combining a large batch in one step requires a way to represent the interactions among its inputs without letting those interactions dominate the fold.

For a quadratic relation, these interactions are mixed terms: each pair of inputs contributes a bilinear expression. Folding one accumulator with  $k$  fresh instances produces  $k$  accumulator–source terms and  $\binom{k}{2}$  source–source terms. Computing and retaining them separately would incur a quadratic dependence on the batch size. Many folding constructions avoid this expansion by first reducing the computation to linear or evaluation claims. We ask how to fold while

retaining the quadratic relation through the challenge that combines the instances.

The companion work [WX26] supplies the algebraic starting point. When the folding coefficients are scaled Cauchy functions of a single challenge, the complete mixed contribution has a polynomial representation with only  $k$  vector coefficients. We call this polynomial the *Cauchy carrier*. CauchyFold turns this representation into a lattice folding protocol.

### 1.1 From the representation to a lattice protocol

A short representation need not be cheap to compute. The carrier’s defining expression still sums over every pair of source states. We give two algorithms that recover its coefficients using fast polynomial arithmetic. One obtains residues at the Cauchy poles using bilinear evaluations; the other evaluates the quadratic map at fixed interpolation points. Both avoid enumerating the source pairs and use quasi-linear arithmetic in  $k$  for fixed relation dimensions.

The lattice commitment imposes a different requirement: its opening must have bounded integer coefficients. The Cauchy coefficients are field elements, so using them to combine lattice openings directly would lose that bound. CauchyFold computes the folded values over an extension field and then encodes them afresh. The new accumulator has the same commitment layout and honest opening bound as its input accumulator. The protocol must then show that this new commitment contains the values computed from the sources and the carrier. A field proof supplies this link and reduces the remaining checks to linear relations over bounded lattice openings.

The extraction proof must also control norms after division by challenge differences. We use modular division to recover candidates and compare them through bounded integer differences, as in [WF26]. For a classical prover that can be restored to its complete state at the specified protocol points, the extractor returns valid source openings or a short nonzero vector in the kernel of a commitment matrix. The theorem gives explicit failure and expected-time bounds for one interactive folding step with a fixed finite sequence of reductions. These outcomes give a computational reduction to the module short integer solution (Module-SIS) problems for the commitment matrices.

A deterministic compiler gives profiles at six selected arities, from 2 to 1024 for a relation consisting of four length-eight dot-product equations. At  $k = 1024$ , the selected profile uses 162.67 KiB of no-retry interactive communication, excluding incoming commitments and the CRS. We give exact statistical and message-length calculations, operation counts separated by type, and lattice-attack estimates for the matrices used by the protocol.

### 1.2 Related work

*Folding and accumulation.* Accumulation separates the task of combining claims from that of finally deciding them [BCMS20,BGH19]. Nova folds committed relaxed R1CS instances after committing a cross term [KST22]. HyperNova and

Protostar broaden the relation interface and support multi-folding [KS24, BC23], while hash-based schemes accumulate code-based claims [BMNW24, BCFW25].

*Lattice folding.* Controlling the witness norm is a central requirement of lattice folding [BC26, BC25]. Neo and Cyclo combine decomposition and range checks with evaluation or principal linear relations [NS26, GLLO26]. Symphony first reduces R1CS statements to linear evaluation claims [Che25], and PikkuFold folds a principal relation carrying a polynomial-evaluation claim [Osa26]. ProtogaLattice also folds a nonlinear relation directly, through a general polynomial interface [BNP26].

CauchyFold retains the quadratic relation until the fold is formed; its field and linear reductions follow that step. Table 1 compares the relations present at the folding challenge scheme by scheme.

**Table 1.** Relation interfaces at the folding challenge.

| Scheme                        | Relation at folding boundary       | Reduction           |   |
|-------------------------------|------------------------------------|---------------------|---|
|                               |                                    | Lattice before fold |   |
| Nova [KST22]                  | Relaxed R1CS                       | ×                   | × |
| HyperNova [KS24]              | CCS evaluation claims              | ×                   | ✓ |
| LatticeFold and Lattice-Fold+ | Linear or evaluation relation      | ✓                   | ✓ |
| Neo [NS26]                    | CCS evaluation claims              | ✓                   | ✓ |
| Cyclo [GLLO26]                | Principal linear relation          | ✓                   | ✓ |
| SALSAA [KLOT25]               | Semi-structured linear relation    | ✓                   | ✓ |
| Symphony [Che25]              | Linear evaluation relation         | ✓                   | ✓ |
| PikkuFold [Osa26]             | Polynomial-evaluation relation     | ✓                   | ✓ |
| ProtogaLattice [BNP26]        | General polynomial relation        | ✓                   | × |
| <b>CauchyFold</b>             | <b>Quadratic residual relation</b> | ✓                   | × |

Reduction before fold means reducing the computation relation to linear or evaluation claims before the challenge that forms the accumulated instance. For SALSAA and PikkuFold, the entry refers to their AIR applications.

*Lattice arguments and extraction.* RPS [KLNO24] gives reductions of knowledge for structured linear and inner-product relations. RnR [KLNO25] uses random projections for structured linear relations. Both are relevant to the linear stage of CauchyFold, while short-challenge invertibility and relaxed binding comparisons support the recovery of bounded openings [LS18, BC25].

Adaptive special soundness studies extraction from an accepting prover beyond a single algebraic transcript [AKLY24], while RoKoko develops rewind-based extraction for lattice protocols [KLN<sup>+</sup>26]. Our comparison step uses the short-kernel method of [WF26]. The carrier representation and its exact retained-width interpretation come from [WX26], under that work’s direct-boundary and support assumptions.

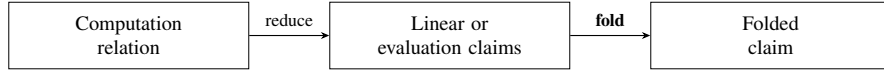
*Field checks and polynomial algorithms.* The field proof uses multilinear extensions and sumcheck [LFKN92], a common route from algebraic constraints to a small number of evaluation claims [Set19,XZZ<sup>+</sup>19,GLS<sup>+</sup>21,DP25]. The polynomial algorithms in [yzGG13,BS05] provide product trees, multipoint evaluation, and interpolation. We apply these routines to the Cauchy mixed term to produce the coefficients that the prover commits.

Section 2 explains the construction, and Section 3 fixes the relations and notation. The carrier algorithms, protocol, and extraction theorem follow in Sections 4–6. Section 7 gives the efficiency bounds and the compiled instance.

## 2 Technical Overview

The main task is to combine quadratic instances without constructing all pairwise mixed terms. Once this is done over a field, the folded values must be committed with bounded lattice openings and linked to the original instances. Figure 1 places these steps in the protocol.

(a) **Linear/evaluation-first route** (*representative*)



(b) **CauchyFold: direct quadratic folding**

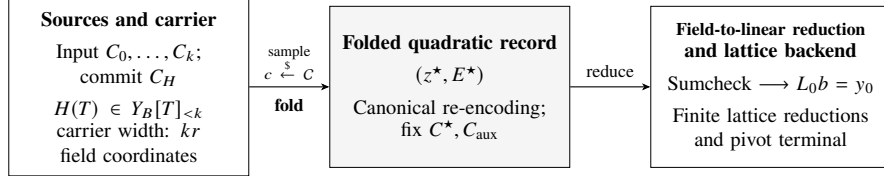


**Fig. 1.** Folding boundary in a representative linear/evaluation-first route and in CauchyFold. The upper row illustrates a representative reduction-before-fold pattern. CauchyFold commits its  $kr$ -coordinate carrier before sampling  $c$ , forms the quadratic fold, and only then reduces the field relation to a scalar linear handoff and finite lattice backend.

### 2.1 Representing the mixed terms

Write the relation as  $Q(z) = E$ , where  $Q$  is a homogeneous quadratic map,  $z$  contains the public and private state, and  $E$  is its residual. Fresh instances have residual zero; an accumulator may carry a nonzero residual from earlier folds. The polarization  $B(x, z) = Q(x + z) - Q(x) - Q(z)$  describes the interaction between two states.

Consider an accumulator  $z_0$  and two fresh states  $z_1, z_2$ . Write  $B_{ij} = B(z_i, z_j)$ , and choose the folding coefficients  $1/T$  and  $1/(T-1)$ . The three mixed terms then have a common denominator:

$$\frac{B_{01}}{T} + \frac{B_{02}}{T-1} + \frac{B_{12}}{T(T-1)} = \frac{(B_{12} - B_{01}) + (B_{01} + B_{02})T}{T(T-1)}.$$

The numerator has just two vector coefficients. These coefficients determine the weighted mixed contribution at every challenge outside  $\{0, 1\}$ , which is precisely the information needed for folding.

For general  $k$ , choose distinct poles  $\xi_i$ , nonzero scales  $\lambda_i$ , and coefficients  $a_i(c) = \lambda_i/(c - \xi_i)$ . The challenge  $c$  is uniform in a set  $\mathcal{C}$  that excludes the poles. Multiplication by  $D(T) = \prod_i (T - \xi_i)$  turns each accumulator–fresh term into a polynomial of degree at most  $k-1$ , and each fresh–fresh term into one of degree at most  $k-2$ . The representation from [WX26] therefore writes the complete mixed contribution as  $H(c)/D(c)$ , where  $\deg H < k$ . We call  $H$  the Cauchy carrier.

Each coefficient lies in  $Y_B$ , the span of the outputs of  $B$ . If this space has dimension  $r$ , representing each coefficient in a fixed basis gives  $kr$  field coordinates. The companion result proves that this width is exact under its stated direct-boundary and support assumptions. Figure 2 illustrates the change in representation.

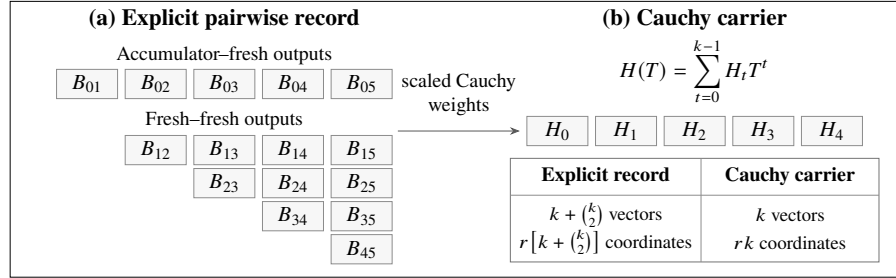


**Fig. 2.** Explicit pairwise record and Cauchy carrier. The direct quadratic expansion contains  $k + \binom{k}{2}$  mixed outputs in  $Y_B$ , whereas the carrier has  $k$  coefficients in  $Y_B$ . If  $r = \dim_{\mathbb{K}} Y_B$ , the corresponding widths are  $r(k + \binom{k}{2})$  and  $rk$  field coordinates. The diagram is schematic for  $k = 5$ ; each cell denotes one element of  $Y_B$ .

The prover commits  $H$  before receiving  $c$ . Otherwise it could choose the carrier to fit only the sampled folded equation. This order will also let the final check certify the source relations, as explained below.

## 2.2 Computing the carrier without the pair table

The defining expression for  $H$  still contains every source pair. The pole-residue algorithm avoids enumerating them by working at the poles. Remove the singular

term  $\lambda_i z_i / (T - \xi_i)$  from the rational folded state and evaluate the remaining regular part at  $\xi_i$ ; call the result  $v_i$ . The residue of the mixed contribution  $H(T)/D(T)$  at  $\xi_i$ , meaning its coefficient of  $(T - \xi_i)^{-1}$ , is

$$R_i = \lambda_i B(z_i, v_i), \quad H(T) = \sum_i R_i \frac{D(T)}{T - \xi_i}.$$

Thus  $k$  evaluations of  $B$  suffice once the  $v_i$  are available. Computing them by separate sums would still visit every pair. Instead, a product tree constructs the common numerator  $N(T) = \sum_i \lambda_i z_i D(T) / (T - \xi_i)$ . Evaluating derivatives of  $N$  and  $D$  at the poles gives all regular parts together. A second tree reconstructs the carrier coefficients from the residues.

An alternative algorithm evaluates  $H$  at  $k$  fixed nonpole points and interpolates. At each point it obtains the mixed contribution from one evaluation of  $Q$ , after subtracting the diagonal source terms. A derivative identity handles their squared denominators. Both constructions use quasi-linear arithmetic in  $k$  for fixed relation dimensions; Section 4 gives their costs and cache requirements.

### 2.3 Committing the folded values with bounded openings

The Cauchy coefficients are extension-field elements. Their integer representatives can be large, so multiplying lattice openings by these coefficients would not preserve the opening bound. We instead compute the folded state and its residual in the field:

$$z^\star = z_0 + \sum_i a_i(c) z_i, \quad E^\star = E_0 + \sum_i a_i(c)^2 E_i + D(c)^{-1} H(c).$$

For valid inputs,  $E_i = Q(z_i)$ , with  $E_i = 0$  for the fresh sources. The prover then encodes  $(z^\star, E^\star)$  canonically and commits that encoding. The output uses the same state commitment matrix, layout, and honest coefficient-norm bound as an accumulated input.

To see why the folded equation checks the sources, first hold their values and  $H$  fixed and put  $\delta_i = Q(z_i) - E_i$ . In the two-source example above, replace  $c$  by a formal variable  $T$  in the folded expressions. Multiplying their relation error  $Q(z^\star) - E^\star$  by  $D(T)^2$  gives a polynomial  $F$  of degree at most four. Evaluating it at the poles gives  $F(0) = \delta_1$  and  $F(1) = \delta_2$ ; its coefficient of  $T^4$  is  $\delta_0$ . If  $F$  is identically zero, all three source relations hold, and the remaining identity forces  $H$  to equal their carrier. Otherwise a random nonpole challenge is a root with probability at most  $4/|\mathcal{C}|$ . For general  $k$ , the same argument gives  $2k/|\mathcal{C}|$ . Section 6.5 justifies fixing these values when commitments may have incompatible openings.

Encoding the output afresh creates a new obligation: the prover must show that the new commitment contains the prescribed fold. A sparse field circuit checks the fold, the output quadratic relation, the public coordinates, and the canonical ranges of the encodings. Sumcheck reduces these constraints to three evaluations of linear forms in the encoded witness. Expanding them in the fixed

field basis and adding the commitment equations gives a public scalar system  $L_0 b = y_0$  over  $\mathbb{F}_q$ . The lattice stage must recover a bounded opening  $b$  for this system. The field checks subsequently establish that the recovered coordinates are canonical bits with the required semantics.

## 2.4 Reducing the linear relation over lattices

Sending the whole opening of  $L_0 b = y_0$  would lose the communication benefit of folding. The lattice protocol instead partitions it into ring blocks and reduces the verification task through a fixed number of layers. Each layer commits the data needed to combine these blocks, checks a projection response, and combines the linear equations by a fresh aggregation challenge. Short ring challenges then form a response, which is split into low and high parts. These parts and the digits of the committed block data form the next layer's witness. Only the final layer sends a bounded response directly.

Both tests use short public randomness descriptions. A seed for Nisan's space-bounded generator [Nis92] produces the ternary projection matrix. Applying the generator to the complete modular norm test bounds the change in acceptance probability for each candidate fixed before the seed. After the projection response is fixed, two extension-field elements generate three aggregation rows. Their joint error is bounded by a polynomial root argument.

## 2.5 Recovering the source openings

The extractor works backwards from a bounded terminal opening. It restores the prover to a saved state and collects five bounded responses on each short-challenge coordinate fiber. Three of these determine the quadratic identity; the full set also bounds comparisons between independent recovery trials. Let  $X_1, X_2$  be the response differences from one center and  $\Delta_1, \Delta_2$  the corresponding challenge differences. Distinct short challenges have invertible differences, so modular division gives two candidates for the witness block. If they differ, the compare-before-clearing method [WF26] forms

$$v = X_1 \Delta_2 - X_2 \Delta_1.$$

This is a nonzero modular kernel of the parent commitment matrix. Its integer norm is controlled by the bounded responses and challenge differences, without requiring a short integer representative of either modular inverse.

When the candidates agree, the extractor still needs to show that the recovered opening is short. It fixes the candidate before sampling a fresh projection test. Agreement with the new accepted responses then links this candidate to the norm check. Inconsistent openings of an unchanged auxiliary commitment instead give a short kernel for that commitment's matrix. Applying these comparisons through the layers yields a bounded opening of the field witness; the field checks and the Cauchy polynomial test then recover the original source relations.

The analysis accounts for the probability and cost of obtaining accepting continuations. It uses fresh trials at each saved state, and a single truncation of the complete extraction gives the finite running-time bound for the Module-SIS reductions.

### 3 Preliminaries and Relations

For a positive integer  $n$ , write  $[n] = \{1, \dots, n\}$  and  $[n]_0 = \{0, \dots, n-1\}$ . The security parameter is  $\kappa$ . We reserve  $\lambda_i$  for the Cauchy scales.

#### 3.1 Fields, rings, and norms

Let  $q = q(\kappa)$  be an odd prime and let  $\mathbb{K}/\mathbb{F}_q$  be an extension of degree  $t$ , with a fixed public basis  $1, \theta, \dots, \theta^{t-1}$ . Semantic relations and the field sumcheck are defined over  $\mathbb{K}$ . Lattice commitments use the rings

$$\mathcal{O} = \mathbb{Z}[X]/(X^d + 1), \quad R_q = \mathcal{O}/q\mathcal{O},$$

where  $d$  is a power of two. The extension-field generator  $\theta$  and the ring variable  $X$  have different roles. In particular, packing field coefficients into  $R_q$  does not identify  $\mathbb{K}$  with  $R_q$ .

For a ring vector  $v$ ,  $\text{coef}(v)$  lists its coefficients in increasing powers of  $X$ , concatenated in vector order. We write  $\text{cen}(v)$  for the lift with coefficients in  $[-(q-1)/2, (q-1)/2]$ , and  $\|v\|_2$  for its Euclidean coefficient norm. An integer vector in  $\mathcal{O}^b$  is measured before reduction modulo  $q$ . For  $c \in \mathcal{O}$ , the operator norm  $\|c\|_{\text{op}}$  is the induced Euclidean norm of negacyclic multiplication, so  $\|cv\|_2 \leq \|c\|_{\text{op}}\|v\|_2$ .

We write  $\text{ct}(f)$  for the constant coefficient of  $f \in R_q$ . Scalar linear forms can be expressed through this operation. For  $a = (a_0, \dots, a_{d-1}) \in \mathbb{F}_q^d$ , put

$$\iota(a) = a_0 - \sum_{j=1}^{d-1} a_j X^{d-j}.$$

Then  $\text{ct}(\iota(a)v) = \langle a, \text{coef}(v) \rangle$ . Applying this identity block by block converts scalar coefficient equations into constant-term equations without changing the underlying scalar relation.

#### 3.2 Quadratic relations and mixed outputs

A relation is specified by an explicit homogeneous quadratic map  $Q : \mathbb{K}^n \rightarrow \mathbb{K}^y$ . Its polarization is the symmetric bilinear map  $B(x, z) = Q(x+z) - Q(x) - Q(z)$ . The description of  $Q$ , rather than a value oracle alone, determines its scalar extension and the field circuit used by the protocol. Write

$$Y_B = \text{span}_{\mathbb{K}}\{B(x, z) : x, z \in \mathbb{K}^n\}, \quad r = \dim_{\mathbb{K}} Y_B.$$

Fix an injective basis map  $U : \mathbb{K}^r \rightarrow \mathbb{K}^y$  with image  $Y_B$ , and its inverse on this image,  $J : Y_B \rightarrow \mathbb{K}^r$ . Thus  $UJb = b$  for every mixed output  $b$ . We distinguish  $y$ , the ambient output dimension, from  $r$ , the dimension needed to represent mixed terms.

A semantic state is  $z = (u_{\text{hom}}, x_{\text{pub}}, z_{\text{priv}})$ , where the first two components are public and  $z_{\text{priv}}$  is private. The coordinate  $u_{\text{hom}}$  homogenizes the original relation. An accumulated record also contains a residual  $E \in \mathbb{K}^y$  and satisfies  $Q(z) = E$ . Fresh records have zero residual and homogenizing coordinate one. For computations over  $\mathbb{F}_q$ , we also require  $z \in \mathbb{F}_q^n$ : all nonconstant extension-field components are zero. The concrete instance uses this base-field condition.

### 3.3 Canonical encodings and commitments

Let  $m_q = \lceil \log_2 q \rceil$ . A field element  $a = \sum_{\nu=0}^{t-1} a_\nu \theta^\nu$  is encoded by the  $m_q$ -bit binary representation of each  $a_\nu \in \{0, \dots, q-1\}$ , in the declared basis order. The map  $\text{Enc}$  concatenates these bits and appends any prescribed zero padding. Decoding is  $\mathbb{F}_q$ -linear on the base-field coordinate vector; groups of  $t$  coefficients are then interpreted in the fixed basis of  $\mathbb{K}$ . The encoding map itself is not linear. If the encoded length is  $L$ , then

$$\text{Dec}(\text{Enc}(s)) = s, \quad \|\text{Enc}(s)\|_2^2 \leq L. \quad (1)$$

Canonicity requires both Boolean bits and a represented integer strictly below  $q$ . Appendix B gives the comparator constraints and the complete segment layout.

The map  $\text{Pack}$  groups consecutive  $d$  encoding bits into each ring element, preserving their coefficient norm. Thus a semantic record in  $\mathbb{K}$  becomes a canonical bit vector, then a ring vector used as the commitment opening. A state commitment has the form

$$C = A_{\text{st}} \text{Pack}(\text{Enc}(z, E)), \quad A_{\text{st}} \in R_q^{a_{\text{st}} \times b_{\text{st}}}.$$

The same matrix is used for the accumulator, all fresh records, and the folded output. Carrier and field-auxiliary encodings use two further matrices  $A_H$  and  $A_{\text{aux}}$ . All independently sampled matrices are uniform, with their joint distribution given by the product of their individual distributions. Additively homomorphic lattice commitments support algebraic manipulation of committed values [BDL<sup>+</sup>16]. Here their binding property is used through the explicit short-kernel problem below.

**Definition 1 (Accumulated and fresh committed relations).** *The public instance is  $(u_{\text{hom}}, x_{\text{pub}}, C)$ . An opening contains  $z_{\text{priv}}$ , a residual  $E$ , and an encoding  $e$ . Put  $z = (u_{\text{hom}}, x_{\text{pub}}, z_{\text{priv}})$ . The opening belongs to  $\mathcal{R}_{\text{acc}}$  if*

$$e = \text{Enc}(z, E), \quad C = A_{\text{st}} \text{Pack}(e), \quad Q(z) = E.$$

*It belongs to  $\mathcal{R}_{\text{fresh}}$  if, in addition,  $u_{\text{hom}} = 1$ ,  $E = 0$ , and  $z \in \mathbb{F}_q^n$ .*

### 3.4 The Cauchy carrier

Fix distinct poles  $\xi_1, \dots, \xi_k \in \mathbb{K}$ , nonzero scales  $\lambda_1, \dots, \lambda_k$ , and an exactly samplable challenge support  $\mathcal{C} \subseteq \mathbb{K} \setminus \{\xi_1, \dots, \xi_k\}$ . Require  $|\mathcal{C}| > 2k$  and a public polynomial bound on the expected cost of exact sampling. Put

$$a_i(c) = \frac{\lambda_i}{c - \xi_i}, \quad D(T) = \prod_{i=1}^k (T - \xi_i), \quad P_i(T) = \frac{D(T)}{T - \xi_i}.$$

The carrier representation from [WX26] is

$$\begin{aligned} H_{\text{src}}(T) &= \sum_{i=1}^k \lambda_i P_i(T) B(z_0, z_i) \\ &\quad + \sum_{1 \leq i < j \leq k} \lambda_i \lambda_j \frac{D(T)}{(T - \xi_i)(T - \xi_j)} B(z_i, z_j). \end{aligned} \tag{2}$$

It lies in  $Y_B[T]_{<k}$ . Its basis coordinates  $JH_{\text{src}}$  therefore consist of  $kr$  field elements. The exact width characterization and its boundary assumptions are given in [WX26].

Given  $H = U\hat{H}$  with  $\hat{H} \in \mathbb{K}^r[T]_{<k}$ , the semantic fold at  $c \in \mathcal{C}$  is

$$\begin{aligned} z^* &= z_0 + \sum_{i=1}^k a_i(c) z_i, \\ E^* &= E_0 + \sum_{i=1}^k a_i(c)^2 E_i + D(c)^{-1} H(c). \end{aligned} \tag{3}$$

For  $H = H_{\text{src}}$  and valid input records, polarization gives  $Q(z^*) = E^*$ . The output is committed through a fresh canonical encoding  $\text{Enc}(z^*, E^*)$ . Its state matrix, encoding length, and honest opening bound are the same as for an accumulated input. No division by  $u^*$  is used; the accumulated relation permits  $u^* = 0$ .

### 3.5 Multilinear extensions and sumcheck

For a table  $f : \{0, 1\}^\ell \rightarrow \mathbb{K}$ , its multilinear extension is the unique polynomial of individual degree at most one agreeing with the table on the Boolean cube. Define

$$\text{eq}(v, b) = \prod_{j=1}^{\ell} (v_j b_j + (1 - v_j)(1 - b_j)).$$

Then  $\tilde{f}(v) = \sum_{b \in \{0, 1\}^\ell} \text{eq}(v, b) f(b)$ .

We use the usual public-coin sumcheck protocol [LFKN92]. At each round the prover fixes a univariate polynomial before the verifier samples the next field challenge. The analysis uses the total-degree identity bound of Schwartz [Sch80].

If the round polynomial has degree at most  $D_{sc}$ , a false claim survives  $\ell$  rounds with probability at most  $D_{sc}\ell/|\mathbb{K}|$ , in addition to any preceding random compression error. Our field check has degree three per round. Together with its initial equality-weighted compression, its error is at most  $4\ell_F/|\mathbb{K}|$ . Section 5.4 expands the terminal evaluations into a scalar linear relation and adds the commitment equations. Sumcheck arguments over modules provide a general framework for connecting polynomial checks to commitment openings [BCS21].

### 3.6 Module-SIS and the extraction model

For a matrix role  $j$ , let  $\vartheta_j = (q, d, a_j, b_j, \beta_j)$ . The corresponding Module-SIS problem gives a uniform matrix  $A_j \in R_q^{a_j \times b_j}$  and asks for

$$v \in \mathcal{O}^{b_j}, \quad A_j v = 0 \pmod{q}, \quad v \bmod q \neq 0, \quad \|v\|_2 \leq \beta_j. \quad (4)$$

Module lattices and short integer relations provide the underlying commitment assumption [Ajt96, LS15]. When  $\beta_j < q$ , the usual nonzero-integer-vector formulation agrees with (4) inside the stated ball: a nonzero vector divisible by  $q$  has norm at least  $q$ .

We denote by  $\text{Adv}_{\vartheta_j}^{\text{MSIS}}(T)$  the maximal success probability within time  $T$  in the chosen computational model. The security theorem keeps a separate advantage term for every independently sampled matrix.

The prover and extractor are classical. A reset restores the complete prover configuration at a specified checkpoint, including its random tape and mutable state. Verifier coins sampled after that checkpoint are fresh in each recovery trial. Continuation success may depend on the challenge. Section 6 accounts for this dependence when sampling further accepting continuations and bounding their cost. Binding enters through comparisons of openings: incompatible openings yield a short kernel for their commitment matrix.

## 4 Cauchy Carrier Algorithms

The formula for  $H_{\text{src}}$  contains  $k + \binom{k}{2}$  bilinear terms. Its coefficients can nevertheless be computed without visiting every pair. The pole-residue algorithm needs only  $k$  evaluations of  $B$ ; an alternative uses  $k$  new evaluations of  $Q$  when the source values are available. Both construct the carrier before the verifier samples its challenge.

### 4.1 The carrier representation

Write  $S = \mathbb{K}^n$  and  $Y = \mathbb{K}^y$ . Throughout this section,  $H$  denotes  $H_{\text{src}}$  from (2). Its committed coordinates are  $\hat{H} = JH \in \mathbb{K}^r[T]_{<k}$ . We use  $P_{ij} = D/((T - \xi_i)(T - \xi_j))$  for the polynomial multiplying the pair  $(i, j)$ .

The same representation gives the rational identity

$$\frac{H(T)}{D(T)} = Q\left(z_0 + \sum_i \frac{\lambda_i z_i}{T - \xi_i}\right) - Q(z_0) - \sum_i \frac{\lambda_i^2 Q(z_i)}{(T - \xi_i)^2}. \quad (5)$$

Both algorithms use the common numerator  $N(T) = \sum_i \lambda_i z_i P_i(T)$ , so that the rational folded source is  $z_0 + N(T)/D(T)$ . All arithmetic takes place over  $\mathbb{K}$ , using the polynomial description of  $Q$  fixed in Section 3.

## 4.2 Pole-residue construction

The pole-residue algorithm works at the poles themselves. It avoids the extra interpolation points and constructs the carrier directly in its image basis. Assume  $\text{char } \mathbb{K} \neq 2$ , as in our instantiation. Define the regular part of the folded source at  $\xi_i$  by

$$v_i = z_0 + \sum_{j \neq i} \frac{\lambda_j z_j}{\xi_i - \xi_j}, \quad R_i = \lambda_i B(z_i, v_i).$$

The regular part  $v_i$  is the constant term after removing  $\lambda_i z_i / (T - \xi_i)$  from  $z_0 + N(T)/D(T)$ . Expanding numerator and denominator at  $\xi_i$  gives the derivative expression below; the residues then reconstruct  $H$ :

$$H(T) = \sum_i R_i P_i(T), \quad v_i = z_0 + \frac{N'(\xi_i)}{D'(\xi_i)} - \lambda_i z_i \frac{D''(\xi_i)}{2D'(\xi_i)}. \quad (6)$$

In particular, the apparent  $k$ -by- $k$  sum in the definition of the  $v_i$  can be evaluated with polynomial arithmetic.

Construct  $D, N$ , differentiate them, and evaluate  $D', D'', N'$  at the poles. The distinctness of the poles gives  $D'(\xi_i) \neq 0$ . Compute their inverses in a batch, form the  $v_i$ , and make  $k$  calls to  $B$ . After applying  $J$  to each  $R_i$ , another numerator tree reconstructs

$$\hat{H}(T) = \sum_i J(R_i) P_i(T).$$

Only  $r$  output coordinates pass through this last tree. In contrast, the first numerator tree processes the  $n$  coordinates of each source vector.

Appendix A derives the regular-part formula and the tree routines.

## 4.3 Multipoint construction

Choose  $k$  distinct public points  $\tau_1, \dots, \tau_k$  outside the pole set. These are internal interpolation points. They are fixed before the protocol and introduce no verifier messages or new challenges. In addition to  $N$ , define  $M_Q(T) = \sum_i \lambda_i^2 Q(z_i) P_i(T)$ . A product tree constructs  $D, N$ , and  $M_Q$  coordinatewise. The recurrence and evaluation routines are given in Appendix A.

The squared denominators in (5) need no second evaluation routine. Formal differentiation gives

$$\sum_i \frac{\lambda_i^2 Q(z_i)}{(T - \xi_i)^2} = \frac{M_Q(T)D'(T) - M'_Q(T)D(T)}{D(T)^2}.$$

Evaluate  $D, D', N, M_Q, M'_Q$  at all  $\tau_j$ . For each point, set  $s_j = z_0 + N(\tau_j)/D(\tau_j)$  and compute

$$h_j = D(\tau_j) \left( Q(s_j) - Q(z_0) - \frac{M_Q(\tau_j)D'(\tau_j) - M'_Q(\tau_j)D(\tau_j)}{D(\tau_j)^2} \right).$$

The values  $h_j$  are evaluations of  $H$ . Interpolating them recovers its  $k$  ambient- $Y$  coefficients; applying  $J$  gives the committed coordinates. Only the  $k$  evaluations  $Q(s_j)$  are new if the source  $Q$ -values are already available. Otherwise those source values require another  $k + 1$  evaluations.

This construction uses standard product trees, remainder trees, and interpolation [vzGG13,BS05]. Appendix A specifies the recurrences and the accounting for inversions.

#### 4.4 Correctness and equivalence

**Theorem 1 (Fast carrier construction).** *For arbitrary source vectors and the data above, the multipoint algorithm outputs the carrier  $\hat{H} = JH$  of (2). In odd characteristic, the pole-residue algorithm outputs the same polynomial. Neither algorithm requires a table of the values  $B(z_i, z_j)$ .*

*Proof.* For multipoint construction, the differentiated numerator identity and (5) give  $h_j = H(\tau_j)$ . There is a unique polynomial of degree below  $k$  with these  $k$  values.

For the pole-residue construction, expand  $\sum_i R_i P_i$ . The terms involving  $z_0$  give the first sum in (2). For a pair  $i < j$ , symmetry of  $B$  combines its two contributions into

$$\lambda_i \lambda_j B(z_i, z_j) \frac{P_i(T) - P_j(T)}{\xi_i - \xi_j} = \lambda_i \lambda_j B(z_i, z_j) P_{ij}(T).$$

The derivative formula computes exactly these  $R_i$ . Both algorithms therefore recover  $H$ , and the linear map  $J$  recovers the same image-basis coefficients.

#### 4.5 Arithmetic complexity

Let  $M(k)$  be a bound on the cost of multiplying two scalar polynomials of degree  $O(k)$  over  $\mathbb{K}$ . Assume  $M(k) \geq k$  and that  $M(k)/k$  is nondecreasing. Also assume  $M(2k) = O(M(k))$ . For brevity, write  $P(k) = M(k) \log(k + 1)$ . Let  $C_Q, C_B, C_J$

be the costs of one  $Q$ ,  $B$ , and  $J$  application. The tree operations cost  $O(P(k))$  per vector coordinate. Hence

$$T_{\text{multi}} = O((n + y)P(k) + kC_Q + kC_J), \quad (7)$$

$$T_{\text{residue}} = O((n + r)P(k) + k(C_B + C_J)). \quad (8)$$

Multipoint interpolation uses the  $y$  ambient output coordinates, whereas residue reconstruction uses the  $r$  image coordinates. Both bounds include the subsequent  $J$  applications. Computing the public basis maps, when necessary, belongs to relation preprocessing.

The large-arity compiler first substitutes the public fresh constants. In each fresh record, 223 base-field coefficients are zero and one is one; the remaining 68 coefficients are private base-field values. Together with the accumulator, output, carrier, and gate values, this leaves  $84k + 728$  free coefficients. A malformed public homogenizing coordinate is rejected. Other public-coordinate constraints remain in the circuit. These substitutions use only the strict fresh format, without assuming the private quadratic equations.

The physical state and carrier segments retain their value-bit layouts. With the specialized comparator, the auxiliary segment has  $192g_K + 48V_K = 3\,696k + 15\,648$  bits, padded to a multiple of 64. At  $k = 1024$ , it occupies 59 381 ring columns and ends with 32 zero bits. The active vector stores the 48 value bits and twelve helpers of each free coefficient, followed by the fixed-one coordinate. Thus

$$N_{\text{used}} = 60(84k + 728) + 1, \quad S_0 = 49(84k + 728) + 1. \quad (9)$$

The full physical vector is  $b_{\text{fix}} + P_{\text{act}}b$ . Its public offset has squared norm  $673k$ : the specialized encodings of zero and one have weights three and four. This offset is subtracted in the commitment equations and cancels in opening comparisons. Accordingly  $S_0$  bounds the active witness. Coordinate insertion preserves differences and introduces no additional commitment matrix.

The remaining semantic equations contribute  $4k + 123$  rows. Combined with the comparisons, this gives

$$M_F = 63(84k + 728) + 4k + 123 = 5\,296k + 45\,987. \quad (10)$$

At  $k = 1024$ , these formulas give  $N_{\text{used}} = 5\,204\,641$ ,  $S_0 = 4\,250\,457$ , and  $\ell_F = 23$ . The initial scalar system has 263 181 rows before backend alignment:  $64 \cdot 4(k + 4)$  commitment equations, twelve field-evaluation equations, and one fixed-one equation.

The streaming reference evaluates each of the  $k + \binom{k}{2}$  bilinear terms, converts it to image coordinates, and accumulates its scalar polynomial coefficient by coefficient. The cubic coordinate cost describes this accumulation routine. It need not store the pair table, although its arithmetic still visits every pair. With  $M(k) = \tilde{O}(k)$  and fixed relation dimensions and map costs, both fast algorithms have  $\tilde{O}(k)$  arity dependence. Section 7 accounts for the remaining protocol costs.

**Table 2.** Carrier algorithms. Map-call counts are for  $k = 16$  and are reported as  $(Q, B, J)$ . Multipoint construction uses cached source  $Q$ -values; without them, add 17 calls to  $Q$ . The streaming reference uses precomputed scalar  $P_i, P_{ij}$ .

| Algorithm       | Polynomial-coordinate work | Map calls, $k = 16$ |
|-----------------|----------------------------|---------------------|
| Streaming pairs | $O(rk^3)$                  | $(0, 136, 136)$     |
| Multipoint      | $O((n + y)P(k))$           | $(16, 0, 16)$       |
| Pole-residue    | $O((n + r)P(k))$           | $(0, 16, 16)$       |

## 5 The CauchyFold Protocol

The protocol has two stages. It first forms the quadratic fold and checks its canonical encoding over  $\mathbb{K}$ . Sumcheck reduces this check to a committed scalar linear system over  $\mathbb{F}_q$ . The lattice stage then reduces the size of its witness until the verifier can check a short response directly. Figure 3 shows the interaction, including which messages are fixed before each challenge.

### 5.1 Front-end parameters

Fix the quadratic map, the public-coordinate map, and the committed relations from Section 3. The input is one accumulator and  $k$  strict fresh records. Their state matrix  $A_{\text{st}}$  is also used for the folded output. Setup independently samples  $A_H$  for the carrier and  $A_{\text{aux}}$  for the field-circuit auxiliary values. Each matrix is uniform in its declared module-matrix space. The encoding layout, segment lengths, public coordinates, and padding are part of the parameters. All backend matrices, dimensions, and challenge distributions are also fixed at setup; we introduce them after the field reduction.

Let  $U : \mathbb{K}^r \rightarrow Y_B$  be the fixed basis embedding for the mixed-output space. The carrier is represented as  $H(T) = U(\widehat{H}(T))$ , where  $\widehat{H}(T) = \sum_{j=0}^{k-1} \widehat{H}_j T^j \in \mathbb{K}^r[T]$ . Thus its encoded semantic payload has exactly  $kr$  extension-field coordinates. Setup also fixes the poles  $\xi_i$ , the nonzero scales  $\lambda_i$ , and the uniformly sampled support  $\mathcal{C} \subseteq \mathbb{K} \setminus \{\xi_1, \dots, \xi_k\}$ . We use  $D(T) = \prod_i (T - \xi_i)$  and  $a_i(c) = \lambda_i / (c - \xi_i)$ .

### 5.2 Commit the carrier before the Cauchy challenge

The input commitments  $C_0, C_1, \dots, C_k$  and their public coordinates are already part of the statement. An honest prover computes the carrier  $H_{\text{src}}$  by either algorithm in Section 4. It sends

$$C_H = A_H \text{Pack}(\text{Enc}(\widehat{H}_0, \dots, \widehat{H}_{k-1})).$$

Only then does the verifier sample  $c$  uniformly from  $\mathcal{C}$ . The fixed coefficient layout enforces  $\deg H < k$ .

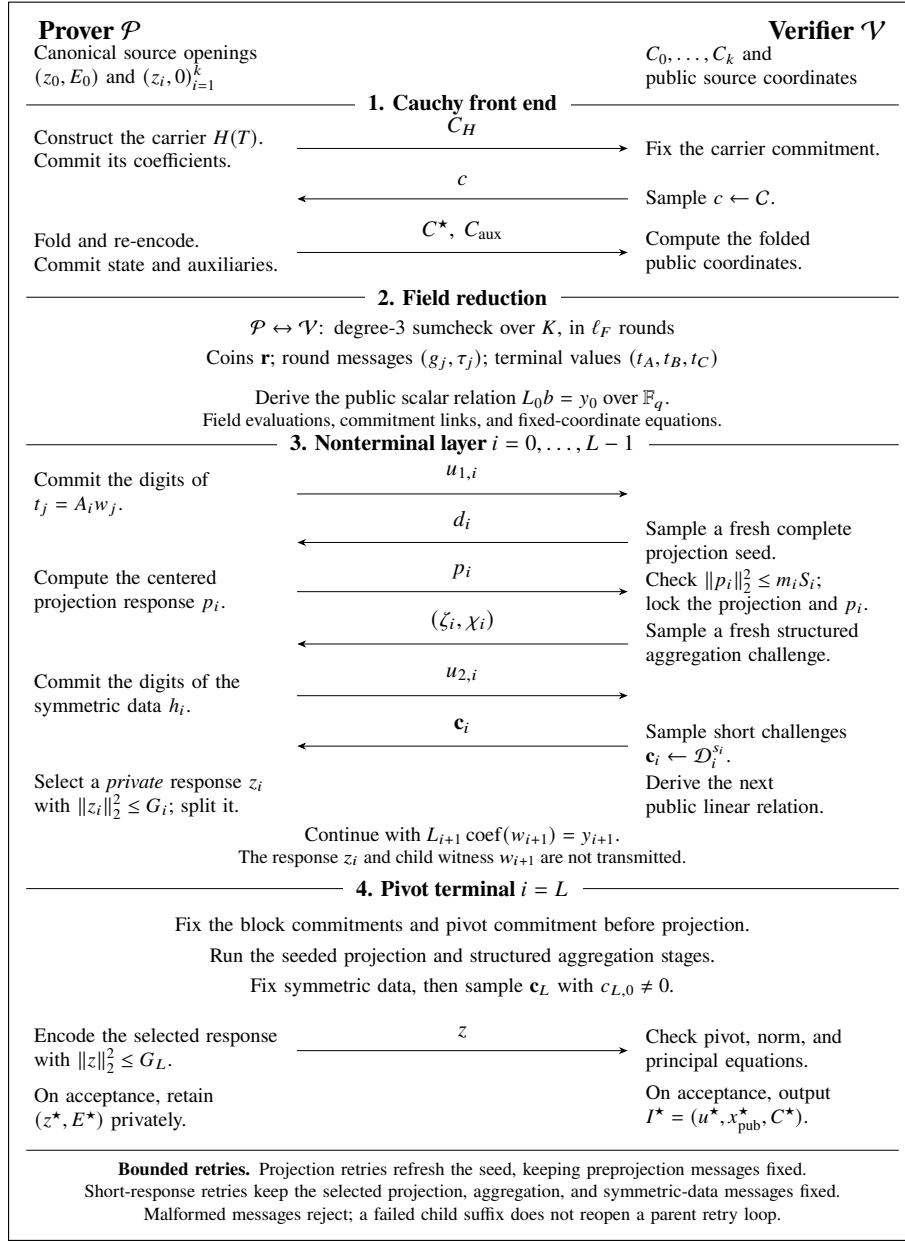


**Fig. 3.** Interaction overview. The carrier commitment is fixed before the Cauchy challenge, and each selected projection response is fixed before the structured aggregation challenge. Nonterminal responses and child witnesses remain private. Section 5 specifies the messages and retry rules; Appendix E gives their canonical serialization.

### 5.3 Form the folded record

The prover computes  $(z^*, E^*)$  by (3). The verifier computes the public coordinates of  $z^*$  by the same linear combination. The prover sends

$$C^* = A_{\text{st}} \text{Pack}(\text{Enc}(z^*, E^*))$$

and the commitment  $C_{\text{aux}}$  to the auxiliary values of the field circuit defined next. Both are fixed before the first field challenge.

### 5.4 Field checks and the scalar linear handoff

The field reduction connects the newly encoded output to the committed sources and carrier. Its public output is a scalar linear system whose witness contains the openings of these commitments and the field auxiliaries. The circuit first checks the following relation on the committed objects: all encodings are Boolean and canonical; padding and public coordinates are correct; each fresh record has homogenizing coordinate one, residual  $E_i = 0$ , and  $z_i \in \mathbb{F}_q^n$ ; equations (3) hold; and  $Q(z^*) = E^*$ . It does not separately evaluate  $Q$  on every input. The carrier fixed before  $c$  makes the folded equation a polynomial test of those input relations, as illustrated in Section 2. Section 6.5 applies that test after comparing the source and carrier openings.

Let  $m_q = \lceil \log_2 q \rceil$  and  $t = \lceil \mathbb{K} : \mathbb{F}_q \rceil$ . Each field value is encoded by  $t$  canonical  $m_q$ -bit coefficients. For every coefficient, the compiler adds the comparison with  $q$  from Appendix B.

The comparison helpers are Boolean variables over  $\mathbb{F}_q$ . Nonlinear gate outputs receive canonical  $\mathbb{K}$  encodings in  $C_{\text{aux}}$ , while affine expressions are lowered directly to linear forms. The resulting sparse matrices satisfy

$$\mathcal{A}b \odot \mathcal{B}b - \mathcal{C}b = 0 \quad \text{over } \mathbb{K}, \quad (11)$$

where  $b \in \mathbb{F}_q^{N_0}$  contains the active encoding coordinates, a fixed-one coordinate, and any alignment padding. Publicly fixed coordinates may be eliminated before forming the matrices. Write the full physical encoding as  $b_{\text{full}} = b_{\text{fix}} + P_{\text{act}}b$ , where  $P_{\text{act}}$  inserts the active coordinates and  $b_{\text{fix}}$  supplies the public constants. The constant coordinate allows all resulting affine expressions to remain linear forms in  $b$ . The template and all dimensions are fixed in advance. The value of  $c$  changes public coefficients, not the shape of the circuit.

Pad the constraint tables with zero rows to  $2^{\ell_F}$  entries. Set  $v_A = \mathcal{A}b$ ,  $v_B = \mathcal{B}b$ , and  $v_C = \mathcal{C}b$ . After all commitments are fixed, the verifier samples  $r \in \mathbb{K}^{\ell_F}$ . The parties run sumcheck on

$$\sum_{v \in \{0,1\}^{\ell_F}} \text{eq}(r, v) (\tilde{v}_A(v) \tilde{v}_B(v) - \tilde{v}_C(v)) = 0. \quad (12)$$

Each round polynomial has degree at most three and is fixed before the next independent uniform challenge. A profile can send all four coefficients, or use the

following two-coefficient encoding. At round  $j$ , write  $g_j(X) = ((1 - r_j) + (2r_j - 1)X)(u_0 + u_1X + u_2X^2)$ , absorbing the equality weight of the preceding challenges into the quadratic factor. For the current claim  $s_j$ , the prover sends  $u_1, u_2$  and the verifier reconstructs  $u_0 = s_j - r_j(u_1 + u_2)$ . This enforces  $g_j(0) + g_j(1) = s_j$  without division, including when  $r_j$  is 0, 1, or  $1/2$ . Every encoded message expands to a degree-three sumcheck message, and the honest polynomial has the required factor. Thus this encoding preserves the original soundness bound. At the final point  $\tau \in \mathbb{K}^{\ell_F}$ , the prover sends  $t_A, t_B, t_C$ . The verifier checks the final value  $\text{eq}(r, \tau)(t_A t_B - t_C)$ .

The remaining claims are

$$\langle \mathcal{A}^\top \text{eq}_\tau, b \rangle = t_A, \quad \langle \mathcal{B}^\top \text{eq}_\tau, b \rangle = t_B, \quad \langle \mathcal{C}^\top \text{eq}_\tau, b \rangle = t_C. \quad (13)$$

Because  $b$  has coordinates in  $\mathbb{F}_q$ , expansion in the fixed field basis gives  $3t$  scalar equations. Adding all segment commitment equations, the constant-coordinate equation, and the zero-padding equations produces the explicit system  $L_0 b = y_0$ . With fixed coordinates eliminated, each segment link is  $C - A \text{Pack}(b_{\text{fix}}) = A \text{Pack}(P_{\text{act}} b)$ , restricted to that segment. The insertion acts on base-field coefficients; ring multiplication is applied after insertion and packing. An honest witness  $b$  consists of bits, so  $S_0 = N_0$  is a valid squared-norm bound; the concrete layout gives a tighter bound by counting fixed coordinates. Pad  $b$  to the first layer's declared length and group its coefficients into ring blocks  $w = \text{Pack}(b)$ . The backend receives the public equation  $L_0 \text{coef}(w) = y_0$  and the opening bound. Its extractor first recovers a bounded coefficient vector. The field comparison in Section 6.5 then establishes Booleanity, canonicity, and the semantic equations.

### 5.5 The recursive linear reduction

A nonterminal layer replaces the current linear system by a child system in a split response and committed digits. The child must satisfy those equations with its prescribed norm bound; the terminal will check a response directly. The block count  $s_i$  below controls this internal reduction and is distinct from the outer folding arity  $k$ .

The lattice backend has layers  $i = 0, \dots, L$ . At layer  $i$ , its input is

$$L_i \text{coef}(w) = y_i, \quad w = (w_0, \dots, w_{s_i-1}), \quad w_j \in R_q^{n_i}. \quad (14)$$

Zero padding makes the coefficient length  $ds_i n_i$ . The honest squared-norm bound is  $S_i$ , while extraction allows  $\gamma_i = \sigma_i \sqrt{S_i}$  with  $\sigma_i \geq 3$ . The public parameters include the main matrix  $A_i \in R_q^{a_i \times n_i}$ , an even radix  $\rho_i < q$ ,  $m_i$  projection rows,  $\tau_i = 3$  aggregation equations, and retry bounds  $R_{P,i}, R_{C,i}$ . The auxiliary matrices used below are independently sampled with their stated dimensions.

The projection family at layer  $i$  is a deterministic map  $\Pi_i : \{0, 1\}^{\ell_i^{\text{seed}}} \rightarrow \{-1, 0, 1\}^{m_i \times ds_i n_i}$ . Its complete seed specifies both the initial block and all hashes of Nisan's generator. The verifier samples all  $\ell_i^{\text{seed}}$  bits independently for each

projection attempt. Appendix C.5 specifies the generator and proves the fixed-vector projection bound used in Section 6.

Each short-challenge support  $\mathcal{D}_i \subset R_q$  is closed under negation and contains zero. Distinct elements have unit differences. Its uniform distribution satisfies

$$|\mathcal{D}_i| = M_i \geq 4, \quad \|\mathbf{c}\|_{\text{op}} \leq T_i^{\text{op}}, \quad \mathbb{E}\|\text{cen}(\mathbf{c}v)\|_2^2 \leq \mu_i \|v\|_2^2. \quad (15)$$

Here  $\mathbf{c}$  is a ring element, unlike the Cauchy challenge  $c \in \mathbb{K}$ . For  $q \equiv 5 \pmod{8}$  and the parameters of Section 7, we use the full coefficient box  $\{-2, -1, 0, 1, 2\}^d$ , with  $M_i = 5^d$  and  $\mu_i = 2d$ . For the concrete degree  $d = 64$ , the spectral calculation in Appendix C.7 gives  $T_i^{\text{op}} \leq 2 \csc(\pi/128) < 82$ , so we use the integer bound  $T_i^{\text{op}} = 82$ . The unit condition follows from the short-difference criterion of [BC25, Lemma 2.4]; the moment calculation is given in Appendix C.

Consider a nonterminal layer and suppress its index where convenient. Put  $\ell_i = \lceil \log_{\rho_i} q \rceil$  and  $b_i^{\text{dig}} = \rho_i/2$ . The deterministic radix decomposition  $\text{Dig}_{\rho_i}$  produces  $\ell_i$  signed digits per coefficient, each of absolute value at most  $b_i^{\text{dig}}$ . Its public linear recomposition map is  $\text{Rec}$ .

*Commitment and projection.* The prover computes  $t_j = A_i w_j$  and forms the digit vector  $\hat{t} = \text{Dig}_{\rho_i}(t_0, \dots, t_{s_i-1})$ . It sends

$$u_1 = B_i \hat{t}, \quad B_i \in R_q^{a_{t,i} \times (\ell_i a_i s_i)}.$$

The verifier sends a fresh complete seed  $d_i \in \{0, 1\}^{\ell_i^{\text{seed}}}$ , which both parties expand to  $\Pi = \Pi_i(d_i)$ . The prover sends a payload-free retry tag or commits to this attempt by sending

$$p = \text{cen}(\Pi \text{coef}(w)), \quad \|p\|_2^2 \leq m_i S_i. \quad (16)$$

There are at most  $R_{P,i}$  attempts, all with the same  $u_1$ . Each retry uses a fresh complete seed. Sending  $p$  fixes the selected projection and response for the rest of this layer. Malformed messages or a violated norm bound cause rejection.

*Aggregation and symmetric data.* After  $p$  is fixed, form  $\tilde{L} = (L_i^\top, \Pi^\top)^\top$  and  $\tilde{y} = (y_i^\top, p^\top)^\top$ . Let  $R_i = \text{rows}(\tilde{L})$  and  $J_i = \lceil R_i/4 \rceil$ . The structured aggregation uses an extension of degree  $t = 4$ . In the fixed basis  $(1, \theta, \theta^2, \theta^3)$ , let  $\pi : \mathbb{K} \rightarrow \mathbb{F}_q^3$  retain the first three coordinates and put

$$\mathcal{H} = \{x_0 + x_1\theta + x_2\theta^2 + \theta^3 : x_0, x_1, x_2 \in \mathbb{F}_q\}.$$

The verifier samples  $\zeta_i \xleftarrow{\$} \mathbb{K}$  and  $\chi_i \xleftarrow{\$} \mathcal{H}$  independently. For  $j = 4b + s < R_i$ , the three aggregation coefficients at position  $j$  are defined by

$$(\alpha_j^{(1)}, \alpha_j^{(2)}, \alpha_j^{(3)}) = \pi(\chi_i \theta^s \zeta_i^b). \quad (17)$$

The message contains the four base-field coordinates of  $\zeta_i$  and the first three coordinates of  $\chi_i$ ; its last coordinate is fixed to one. It therefore occupies seven

base-field elements. Set  $v^{(e)} = \alpha^{(e)}\tilde{L}$  and  $b_e = \alpha^{(e)}\tilde{y}$ . Apply the constant-term lifting  $\iota$  from Section 3 blockwise to these coefficient rows. This gives public vectors  $\phi_j^{(e)} \in R_q^{n_i}$  such that

$$v^{(e)} \text{coef}(w) = \text{ct} \sum_j \langle \phi_j^{(e)}, w_j \rangle. \quad (18)$$

To express the principal equation for a linear combination of blocks, the prover forms the upper-triangular symmetric data

$$h_{ab}^{(e)} = \frac{1}{2} (\langle \phi_a^{(e)}, w_b \rangle + \langle \phi_b^{(e)}, w_a \rangle), \quad 0 \leq a \leq b < s_i.$$

These are the coefficients of the quadratic identity in the short challenges. The prover commits to their radix digits as  $u_2 = D_i \hat{h}$ , with  $D_i \in R_q^{a_{h,i} \times (\ell_i \tau_i s_i (s_i+1)/2)}$ . The scalar diagonal identities are  $\text{ct} \sum_j h_{jj}^{(e)} = b_e$ .

*Short response and child.* Only after  $u_2$  is fixed does the verifier sample independent  $\mathbf{c}_j \leftarrow \mathcal{D}_i$ . The honest response is  $z = \text{cen}(\sum_j \mathbf{c}_j w_j)$ . The prover may request at most  $R_{C,i} - 1$  further short challenges to obtain  $\|z\|_2^2 \leq G_i$ , where the integer threshold satisfies

$$G_i + 1 \geq 2\mu_i S_i. \quad (19)$$

Thus the smallest threshold allowed by this analysis is  $G_i = \lceil 2\mu_i S_i \rceil - 1$ . These retries keep  $u_1, \Pi, p, \zeta_i, \chi_i, u_2$  unchanged.

Once a response is selected, write  $z = z_{\text{lo}} + \rho_i z_{\text{hi}}$  over the integers, with  $\|z_{\text{lo}}\|_\infty \leq b_i^{\text{dig}}$ . The child witness is  $x = (z_{\text{lo}}, z_{\text{hi}}, \hat{t}, \hat{h})$ . Its public scalar linear system is obtained by expanding

$$\begin{aligned} B_i \hat{t} &= u_1, & D_i \hat{h} &= u_2, \\ A_i(z_{\text{lo}} + \rho_i z_{\text{hi}}) &= \sum_j \mathbf{c}_j \text{Rec}(\hat{t}_j), \end{aligned} \quad (20)$$

$$\left\langle \sum_j \mathbf{c}_j \phi_j^{(e)}, z_{\text{lo}} + \rho_i z_{\text{hi}} \right\rangle = \sum_{a,b} \mathbf{c}_a \mathbf{c}_b \text{Rec}(\hat{h}_{ab}^{(e)}), \quad (21)$$

$$\text{ct} \sum_j \text{Rec}(\hat{h}_{jj}^{(e)}) = b_e. \quad (22)$$

In an upper-triangular representation, each off-diagonal term in (21) has coefficient  $2\mathbf{c}_a \mathbf{c}_b$ . All coefficients are now public, so these equations are linear in  $x$ . Any padding added for the next block partition receives an explicit zero equation. The next layer proves this system with its prescribed norm bound. After the child is submitted, a rejecting suffix ends the attempt; it does not reopen the parent's response retry loop.

The child has raw coefficient length

$$N_{i+1}^{\text{raw}} = 2dn_i + d\ell_i \left( a_i s_i + \tau_i \frac{s_i(s_i+1)}{2} \right). \quad (23)$$

Its honest energy bound combines the split-response bound with the energies of the two digit segments. Equation (53) in Appendix C.8 gives the bound used by the parameter calculation. Zero padding increases only the length.

### 5.6 The pivot terminal

The terminal creates no child witness. It sends the bounded response and symmetric data directly, while one block commitment is reconstructed from the response. A separate pivot commitment fixes the digit opening of that block before projection. Let its main matrix be  $A_L \in R_q^{a_L \times n_L}$  and put  $t_j = A_L w_j$ . Fix an even pivot radix  $\rho_p < q$ , let  $\ell_p = \lceil \log_{\rho_p} q \rceil$ , and sample an independent  $B_p \in R_q^{a_p \times (a_L \ell_p)}$ .

Before projection, the prover sends  $t_1, \dots, t_{s_L-1}$  and  $v_p = B_p \text{Dig}_{\rho_p}(t_0)$ . Projection and aggregation follow the preceding layer definition. The prover then sends the symmetric values  $h_{ab}^{(e)}$  before the short challenge. For each  $e$ , it omits only  $\text{ct}(h_{00}^{(e)})$ , which the verifier reconstructs as  $b_e - \sum_{j \geq 1} \text{ct}(h_{jj}^{(e)})$ . All other coefficients of  $h_{00}^{(e)}$  are transmitted.

The verifier samples  $\mathbf{c}_0 \leftarrow \mathcal{D}_L \setminus \{0\}$  and  $\mathbf{c}_j \leftarrow \mathcal{D}_L$  for  $j \geq 1$ . It uses the same fixed-prefix retry rule. On receiving  $z$ , it sets

$$t_0(\mathbf{c}, z) = \mathbf{c}_0^{-1} \left( A_L z - \sum_{j=1}^{s_L-1} \mathbf{c}_j t_j \right). \quad (24)$$

The inverse exists by the unit-difference condition on short challenges. The verifier checks

$$B_p \text{Dig}_{\rho_p}(t_0(\mathbf{c}, z)) = v_p, \quad \|z\|_2^2 \leq G_L,$$

and every principal equation (21), with  $z$  in place of its split representation. The response threshold is

$$G_L + 1 \geq 2\mu_L \frac{M_L}{M_L - 1} S_L. \quad (25)$$

Equivalently, the smallest integer threshold is  $G_L = \lceil 2\mu_L M_L S_L / (M_L - 1) \rceil - 1$ . The factor accounts for excluding the zero challenge in coordinate zero. The terminal uses only the two matrices  $A_L$  and  $B_p$ .

Projection responses and the terminal response use a canonical, fixed-padded signed Rice encoding of their centered integer coefficients. The decoder checks canonicity and the integer norm before mapping the coefficients into  $\mathbb{F}_q$ . Retry tags have no payload; malformed payloads reject. Appendix E specifies the byte grammar.

## 6 Security

The proof follows the protocol in reverse. Starting with the terminal response, the extractor recovers a bounded opening of each preceding linear relation. The

field step then establishes that the initial opening is a canonical encoding, and the Cauchy step recovers the source quadratic relations. An inconsistent comparison instead returns a short nonzero kernel of the matrix whose commitment is being compared.

Within a lattice layer, five accepted continuations on each short-challenge coordinate fiber give a modular witness candidate. Any three distinct values determine the quadratic identity, while all five are retained for comparisons across independent trials. A separate trial, with fresh projection and aggregation randomness, certifies its norm and the original linear equations. The proof restores the prover only to complete saved states: a short-challenge reset fixes all earlier messages, whereas a projection reset fixes only the commitments sent before projection. The source-opening steps use the same distinction at the field and Cauchy challenges.

### 6.1 A finite extract-or-short-kernel theorem

Fix the setup and a complete pre-Cauchy checkpoint. Each recovery experiment restores its own complete checkpoint and samples fresh continuation coins. Local resets preserve all messages and challenges fixed before that checkpoint.

Three errors contribute at layer  $i$ . A vector of norm greater than  $\gamma_i$  passes one seeded projection with probability at most  $\epsilon_{P,i}$ . This fixed-vector bound follows by expressing the norm test as a bounded-space computation and applying Lemma 7. Across  $R_{P,i}$  attempts, use  $\hat{\epsilon}_{P,i} = 1 - (1 - \epsilon_{P,i})^{R_{P,i}}$ . A fixed nonzero residual survives structured aggregation with probability at most  $\epsilon_{\text{agg},i}$ , as given by Lemma 8. Finally, coordinate recovery loses at most  $10R_{C,i} \sum_j M_{ij}^{-1}$ . Thus the local loss is

$$\epsilon_i = \hat{\epsilon}_{P,i} + \epsilon_{\text{agg},i} + 10R_{C,i} \sum_{j=0}^{s_i-1} \frac{1}{M_{ij}}, \quad (26)$$

where  $M_{ij} = M_i$  except that  $M_{L0} = M_L - 1$  at the pivot terminal. Let  $\epsilon_F \leq 4\ell_F/|\mathbb{K}|$  be the error of the field check. Adding the Cauchy consistency error gives

$$\kappa_{\text{alg}} = \epsilon_F + \frac{2k}{|\mathcal{C}|} + \sum_{i=0}^L \epsilon_i. \quad (27)$$

Appendix C.10 gives the exact expressions and the projection specialization used for the concrete profiles.

The specified main and auxiliary kernel radii at a nonterminal layer are

$$\beta_{A_i} = \left\lceil 3\sqrt{3} T_i^{\text{op}} \sqrt{1 + \rho_i^2} \gamma_{i+1} \right\rceil, \quad \beta_{B_i} = \beta_{D_i} = \lceil 2\gamma_{i+1} \rceil. \quad (28)$$

At the terminal they are

$$\beta_{A_L} = \lceil 3\sqrt{3} T_L^{\text{op}} \sqrt{G_L} \rceil, \quad \beta_{B_p} = \left\lceil \rho_p \sqrt{da_L \ell_p} \right\rceil. \quad (29)$$

Each of  $A_{\text{st}}, A_H, A_{\text{aux}}$  has comparison radius  $\lceil 2\gamma_0 \rceil$ . These radii use the bound supplied by the linear-chain extractor; canonicity is established only in the subsequent field step.

**Theorem 2 (Finite-node extraction).** *Use the seeded-projection protocol in Section 5. Assume the field, encoding, and challenge conditions there, including  $M_{ij} \geq 5$ ;  $q \geq 128\gamma_i$  and the stated independent-row projection bound at every layer; and the child length and energy conditions (23) and (53). The 384-row profiles satisfy the stronger condition  $q \geq 176\gamma_i$ . For every fixed pre-Cauchy checkpoint  $s$  of a classical prover, let  $p(s)$  be the acceptance probability of the actual continuation. There is a resettable extractor that returns one of the following with probability at least  $\max\{0, p(s) - \kappa_{\text{alg}}\}$ :*

1. *canonical openings of all source commitments, satisfying their specified accumulator or strict fresh relations, together with the carrier consistency equation and an output opening on one actual accepting Cauchy replay;*
2. *a specified matrix  $A_j$  and an integer ring vector  $v$  such that  $A_j v = 0 \bmod q$ ,  $v \bmod q \neq 0$ , and  $\|v\|_2 \leq \beta_j$ .*

*For a prover whose continuations have the required uniform conditional cost bounds, the extractor has the expected-time recurrences in Appendix C.9. For fixed internal depth and polynomial-size parameters, these bounds are polynomial. Honest execution fails with probability at most*

$$\epsilon_{\text{comp}} \leq \sum_{i=0}^L (2^{-R_{P,i}} + 2^{-R_{C,i}}). \quad (30)$$

*In the selected profiles  $R_{P,i} = R_{C,i} = 160$ , so this is  $2(L+1)2^{-160}$ .*

The following argument derives the recovery and comparison guarantees; Appendix C supplies the sampling, projection, and norm lemmas.

## 6.2 Local recovery from five coordinate challenges

We first recover algebraic candidates, following the reduction-of-knowledge viewpoint [KP23], and certify their norms in the next step. Fix a short-challenge checkpoint, including  $u_1, u_2$ , the chosen projection, and the aggregation rows. A successful child extraction supplies the child linear equations and an opening with norm at most  $\gamma_{i+1}$ . Recomposition therefore bounds its response by

$$\|z\|_2 \leq Z_i^{\text{ext}} := \sqrt{1 + \rho_i^2 \gamma_{i+1}}. \quad (31)$$

This is the bound used in extraction, rather than the honest threshold  $\sqrt{G_i}$ . The terminal directly enforces  $Z_L^{\text{ext}} = \sqrt{G_L}$ .

Retain one successful center  $(\mathbf{c}, z)$ . For coordinate  $j$ , seek four further successful continuations that change only  $\mathbf{c}_j$ , using distinct values. After  $r \in \{1, 2, 3, 4\}$  values have been excluded, a failed call stops the search with probability  $r/M_{ij}$ ,

as in Lemma 4. If their auxiliary openings disagree modulo  $q$  with the center, subtract the two openings of  $u_1$  or  $u_2$  and return a  $B_i$ - or  $D_i$ -kernel. Otherwise  $t$  and  $h$  are fixed across all five continuations. Write

$$\Delta_b = \mathbf{c}_j^{(b)} - \mathbf{c}_j, \quad X_b = z^{(b)} - z, \quad w_j^{(b)} = X_b \Delta_b^{-1}, \quad b \in \{1, 2, 3, 4\}.$$

The commitment equation gives  $A_i X_b = t_j \Delta_b$ . If two recovered slopes differ, relabel them as 1, 2. The integer vector

$$v = X_1 \Delta_2 - X_2 \Delta_1 \tag{32}$$

is a nonzero modular kernel. Its norm is at most  $3\sqrt{3} T_i^{\text{op}} Z_i^{\text{ext}}$ . Only short numerators and short challenge differences are lifted to the integers. The modular inverse used to define a slope need not have a short integer lift. This is the compare-before-clearing principle of [WF26] in the present interface.

If all four slopes agree, call the result  $w_j$  and retain the five response-challenge pairs. Replace the  $j$ th challenge by a formal variable  $T$  in the principal equation, and replace the response by  $z + (T - \mathbf{c}_j)w_j$ . The resulting polynomial has degree at most two and vanishes at all five chosen values. Any three have pairwise unit differences, so its Vandermonde matrix is invertible over  $R_q$ . The quadratic coefficient gives

$$\langle \phi_j^{(e)}, w_j \rangle = h_{jj}^{(e)}, \quad A_i w_j = t_j.$$

Summing the diagonal identities recovers the scalar aggregation equations.

The coordinate-sampling analysis averages over the original uniform center, allowing continuation success to depend on the challenge (Lemma 4). It loses at most  $10 \sum_j M_{ij}^{-1}$  success mass and makes at most  $1 + 4s_i$  oracle calls in expectation. Accounting for the bounded response attempts gives  $10R_{C,i} \sum_j M_{ij}^{-1}$  in (26). At the pivot terminal, disagreement in reconstructed  $t_0$  instead produces a  $B_p$ -kernel: its canonical digit openings have length  $da_L \ell_p$  and differ coefficient-wise by at most  $\rho_p$ . If they agree, the same five-point recovery applies.

### 6.3 Certifying a recovered candidate

The slopes just recovered satisfy the sampled aggregation equations. They need not yet satisfy the original linear system or its norm bound. To certify them, return to the complete checkpoint before projection, where  $u_1$  is fixed, and run a fresh recovery trial.

For a fixed candidate  $w$  with  $L_i \text{coef}(w) \neq y_i$ , the augmented residual is nonzero for every projection response. The structured aggregation challenge is sampled only after that response is fixed, so Lemma 8 bounds the probability that all three derived rows miss it by  $\epsilon_{\text{agg},i}$ . If  $w$  satisfies the linear system but  $\|w\|_2 > \gamma_i$ , an incorrect projection response is covered by the same aggregation bound. A correct projection response must pass the fixed-vector test. Its probability over at most  $R_{P,i}$  fresh complete seeds is at most  $\hat{\epsilon}_{P,i}$ . Thus a fresh recovery returns any specified bad  $w$  with probability at most

$$\delta_i^{\text{ret}} = \epsilon_{\text{agg},i} + \hat{\epsilon}_{P,i}. \tag{33}$$

The comparison respects commitment lifetimes. Across new projection trials,  $u_1$  remains fixed. Different  $\hat{t}$  openings therefore give a  $B_i$ -kernel. Equal  $\hat{t}$  but different recovered slopes give a nonzero kernel by Lemma 5. It compares all pairs of edges from the two five-point fibers and returns one with norm at most  $5T_i^{\text{op}}Z_i^{\text{ext}}$ . Since  $5 < 3\sqrt{3}$ , this fits the radii in (28) and (29). The commitment  $u_2$  may change after a new projection and aggregation, so its  $\hat{h}$  openings are not compared across that boundary. At the terminal, the fixed pivot commitment replaces the  $u_1$  comparison.

An accepted trial has a biased challenge distribution, so the fixed-candidate bound must be applied before conditioning on acceptance. Run one recovery trial. On a successful candidate, independently repeat the same recovery trial until it returns a candidate or kernel. Incompatible candidates yield a kernel. Compatible candidates are accepted only after their original linear relation and norm bound are checked. Lemma 3 shows that this procedure loses at most  $\delta_i^{\text{ret}}$  success probability and costs at most twice the expected cost of one recovery trial, plus comparison and checking.

#### 6.4 Finite composition and acceptance conditioning

Put  $\Lambda_i = \sum_{h=i}^L \epsilon_h$ . The terminal supplies the base case by direct verification. Suppose the child extractor succeeds with probability at least  $p_t - \Lambda_{i+1}$  at every fixed child checkpoint  $t$ . An actual forward execution submits at most one child proposal: once submitted, a suffix rejection cannot trigger another parent retry. Averaging over the chosen checkpoint therefore subtracts  $\Lambda_{i+1}$  only once. Coordinate recovery then uses the actual success probabilities of this oracle.

Combining local recovery with (33) gives success at least  $p_i - \Lambda_i$  at layer  $i$ . The induction preserves the witness relation, its extracted norm, the role and radius of every kernel, and one actual accepting continuation from the center branch. This last record identifies the output commitment and challenge to which the final witness belongs.

The same induction gives (54) by applying the uniform conditional cost bound at every reached checkpoint. Appendix C.9 justifies summing costs when the number of invocations depends on earlier outcomes. For growing depth, the product  $\prod_i 2R_{C,i}(1 + 4s_i)$  must also be controlled; the theorem asserts polynomial expected time for fixed depth.

#### 6.5 From linear openings to source witnesses

Fix the post-Cauchy commitments. One raw field trial samples fresh field coins, runs sumcheck, and invokes the linear-chain extractor. It returns a kernel or an opening  $b$  of  $L_0b = y_0$  with  $\|b\|_2 \leq \gamma_0$ . For any fixed  $b$  violating (11), the random equality test misses its nonzero residual with conditional hazard at most  $|\mathbb{K}|^{-1}$  in each of its  $\ell_F$  restrictions. Conditional on a nonzero initial claim, each degree-three sumcheck round has hazard at most  $3|\mathbb{K}|^{-1}$  [LFKN92]. Multiplying the conditional survival probabilities bounds their combined failure by  $\epsilon_F$ . The

terminal linear equations bind the final three evaluations. Thus a fixed bad opening returns with probability at most  $\epsilon_F$ .

Apply the preceding independent-trial comparison at this checkpoint. If the openings differ, some committed segment differs: the uncommitted constant and padding coordinates are already fixed by  $L_0b = y_0$ . The difference of the two segment openings is a kernel for its corresponding matrix. Public offsets cancel, and coordinate insertion followed by restriction to a segment has Euclidean operator norm at most one. Its radius is therefore at most  $2\gamma_0$ , including when fixed coordinates have been eliminated. If they agree, check the full field relation and decoding directly. This gives canonical source, carrier, output, and auxiliary values with the post-Cauchy semantics.

One more comparison moves to the pre-Cauchy checkpoint. Across its independent trials, only source and carrier openings are compared; output and auxiliary commitments are allowed to depend on  $c$ . Incompatible source or carrier openings give an  $A_{\text{st}}$ - or  $A_H$ -kernel. For a fixed compatible tuple, put  $\delta_i = Q(z_i) - E_i$  and  $P_i(T) = D(T)/(T - \xi_i)$ . Its accepted fold implies that  $c$  is a root of

$$F(T) = D(T)^2\delta_0 + \sum_{i=1}^k \lambda_i^2 P_i(T)^2 \delta_i + D(T)(H_{\text{src}}(T) - H(T)). \quad (34)$$

The polynomial is identically zero exactly when every source relation holds and  $H = H_{\text{src}}$ . Indeed, evaluation at  $\xi_i$  isolates  $\delta_i$  for  $i \geq 1$ . Once these vanish, the coefficient of  $T^{2k}$  isolates  $\delta_0$ . The remaining identity forces  $H = H_{\text{src}}$ . Otherwise a nonzero scalar coordinate has degree at most  $2k$ , giving a return probability at most  $2k/|\mathcal{C}|$ . The final comparison proves Theorem 2.

## 6.6 Averaging over setup and reducing each role

Let  $A_1, \dots, A_J$  be the complete set of independently sampled matrices. Repeated uses of  $A_{\text{st}}$  constitute one role. Associate  $\vartheta_j = (q, d, a_j, b_j, \beta_j)$  with coefficient  $\ell_2$  Module-SIS and the nonzero-modulo- $q$  condition of Theorem 2. Let  $\bar{p}_N$  be the seeded protocol's actual acceptance probability averaged over setup and adversarial initialization, and let  $P_N$  bound the extractor's entire expected running time under the same distribution.

**Theorem 3 (Role-specific computational reduction).** *Assume the product setup above and the hypotheses of Theorem 2. For a global time budget  $B_N$ , truncate the complete extractor once after  $B_N$  steps, obtaining  $E_{B_N}$ . For each role  $j$ , let  $S_{j,N}^{\max}$  bound all extra work needed to embed a Module-SIS challenge, generate the other setup coordinates, initialize the prover, and return a checked kernel. In a computation model with exact uniform finite-alphabet sampling,*

$$\begin{aligned} & \Pr[E_{B_N} \text{ returns source witnesses}] \\ & \geq \max \left\{ 0, \bar{p}_N - \kappa_{\text{alg}} - \frac{P_N}{B_N} - \sum_{j=1}^J \text{Adv}_{\vartheta_j}^{\text{MSIS}}(B_N + S_{j,N}^{\max}) \right\}. \end{aligned} \quad (35)$$

For bounded implementations of setup sampling from unbiased bits, subtract additionally  $\sum_j \eta_{\text{set},j}$ , where  $\eta_{\text{set},j}$  is the declared sampling-failure bound for the extra setup in reduction  $j$ .

*Proof.* Averaging Theorem 2 over setup and initialization gives success at least  $\bar{p}_N - \kappa_{\text{alg}}$ . Markov's inequality charges  $P_N/B_N$  for truncating the complete extractor. For each  $j$ , embed the external Module-SIS matrix in role  $j$  and sample every other matrix independently. This preserves the product-setup distribution, and a role- $j$  kernel solves the embedded instance within time  $B_N + S_{j,N}^{\max}$ . Sum over the disjoint kernel-output events. Bounded setup sampling adds its failure probability by coupling with exact sampling.

Appendix D specifies the setup and running-time costs of these reductions, including bounded sampling from unbiased bits.

## 7 Efficiency and Concrete Instantiation

We analyze carrier construction, field reduction, and the lattice backend, then report communication and arithmetic counts for six selected arities. Communication follows the canonical serialization in Appendix E. These are parameter-derived costs; no end-to-end runtime benchmark is reported.

### 7.1 Asymptotic cost

Let  $N = N_0^{\text{raw}}$  be the initial coefficient-witness length. Write  $W_F$  for construction and application of the sparse field relation, including the transposes used for the final sumcheck evaluations, and let  $C_H$  be the carrier cost from Theorem 1. At layer  $i$ , the witness is padded to  $N_i = ds_i n_i$  coefficients. Its child has raw length

$$N_{i+1}^{\text{raw}} = \frac{2N_i}{s_i} + dl_i \left( a_i s_i + \tau_i \frac{s_i(s_i + 1)}{2} \right). \quad (36)$$

The first term is the split response; the second contains commitment digits and symmetric data. This is equation (23) expressed in terms of the padded length  $N_i$ .

One nonterminal layer performs

$$O(a_i s_i n_i + \tau_i s_i^2 n_i + a_{t,i} \ell_i a_i s_i + a_{h,i} \ell_i \tau_i s_i^2)$$

ring operations, in addition to projection, aggregation, and public-transpose work. The public maps can be applied in either direction by streaming their coefficients.

**Theorem 4 (Finite-schedule efficiency).** Fix  $0 < \nu \leq 1/3$  and a constant number  $L$  of nonterminal layers. Choose

$$s_i = \max\{3, \lceil (N_i^{\text{raw}})^\nu \rceil\}, \quad n_i = \left\lceil \frac{N_i^{\text{raw}}}{ds_i} \right\rceil,$$

and use a terminal block count of the same order. Suppose the resulting schedule satisfies the norm, modulus, challenge, and statistical conditions of Section 6. If the remaining public dimensions and retry caps are polynomial in the security parameter and  $\log q = \text{poly}(\kappa) + O(\log N)$ , then, for  $\gamma = (1 - \nu)^{L+1}$ ,

$$T_P = \tilde{O}_\kappa(W_F + C_H + N^{1+\nu}), \quad B_{P \rightarrow V}^{\text{fold}} = \tilde{O}_\kappa(N^\gamma + 1), \quad (37)$$

$$T_V = \tilde{O}_\kappa(W_F + N), \quad B_{V \rightarrow P} = \tilde{O}_\kappa(N^\nu + 1). \quad (38)$$

The notation hides polynomial security-parameter factors and logarithms, but no powers of  $k$ ,  $s_i$ , or  $N$ . Incoming fresh commitments add  $O(ka_{\text{std}} \log q)$  bits.

The proof is given in Appendix E.1.

## 7.2 Compiled relation and profile families

The relation consists of four dot-product equations,

$$Q_j(z) = \sum_{\ell=0}^7 a_{j,\ell} b_{j,\ell} - u x_j, \quad 0 \leq j < 4.$$

The 69 coordinates of  $z$  are  $u$ , the four public values  $x_j$ , and the 64 private values  $a_{j,\ell}, b_{j,\ell}$ . Fresh records have  $u = 1$ , zero residual, and base-field coordinates, so the dot products must equal the stated  $x_j$ . This sparse relation serves as a concrete instance of the protocol.

The four polarized outputs span  $Y_B = Y = \mathbb{K}^4$ , so  $n = 69$  and  $y = r = 4$ . At arity  $k$ , the carrier contains  $4k$  extension-field values. The profiles with  $k \leq 32$  use the reference prefix comparator. For  $k = 1024$ , the compiler eliminates publicly fixed fresh coordinates and uses a comparator specialized to  $q = 2^{48} - 59$ . Its active length is  $N_{\text{used}} = 5040k + 43681$ , its field-row count is  $M_F = 5296k + 45987$ , and its honest energy bound is  $S_0 = 4116k + 35673$ . The field tables are padded to  $2^{\ell_F}$  rows. Appendix B derives the segment layout and constraints.

All profiles use

$$q = 2^{48} - 59, \quad \mathbb{K} = \mathbb{F}_q[\theta]/(\theta^4 - 4\theta^2 + 2), \quad R_q = \mathbb{F}_q[X]/(X^{64} + 1).$$

They also share the short-challenge set, canonical value encoding, retry caps, and pivot parameters. The large-arity profile uses the specialized helpers, two-coefficient sumcheck messages, and exact Rice capacities described in Appendices B and E; the five smaller profiles retain the reference helper and message formats. Every modular projection has 384 rows, with squared norm slack  $\sigma^2 = 38400/4121$ . The main, front-end, auxiliary, and pivot row ranks are respectively 16, 4, 6, and 2. Full parameters and schedules appear in Appendix D.7.

At  $k = 1024$ , the carrier has 4096 extension-field values. Eliminating fixed coordinates and specializing the canonical comparison gives 5204641 active coefficient positions and 5469091 field rows. The source commitment format and folding relation remain the same. The selected schedules give no-retry interaction below 163 KiB at all six reported arities.

**Table 3.** Internal size and no-retry interaction of the selected arity profiles.

| $k$                     | Handoff<br>coefficients | Field<br>constraints | $\ell_F$  | Interaction<br>(bytes) | Interaction<br>(KiB) |
|-------------------------|-------------------------|----------------------|-----------|------------------------|----------------------|
| 2                       | 129 025                 | 200 813              | 18        | 74 735                 | 72.98                |
| 4                       | 188 161                 | 293 031              | 19        | 92 032                 | 89.88                |
| 8                       | 306 433                 | 477 467              | 19        | 116 298                | 113.57               |
| 16                      | 542 977                 | 846 339              | 20        | 156 322                | 152.66               |
| 32                      | 1 016 065               | 1 584 083            | 21        | 154 820                | 151.19               |
| <b>1024<sup>†</sup></b> | <b>5 204 641</b>        | <b>5 469 091</b>     | <b>23</b> | <b>166 579</b>         | <b>162.67</b>        |

The interaction columns give exact bidirectional syntax counts without retries. They exclude incoming commitments, the CRS, and application-level public-input envelopes. Handoff counts exclude partition-alignment padding, and field constraints are unpadding. Each row uses the block partition and radix schedule specified in Appendix D.7. <sup>†</sup> The  $k = 1024$  row uses the specialized compiler and compressed message formats, with four nonterminal layers. All rows use the common matrix ranks and 384 projection rows.

The profiles with  $k \leq 16$  enter the pivot terminal directly. The  $k = 32$  profile uses two nonterminal layers and splits its auxiliary commitment into two independently sampled matrices. The  $k = 1024$  profile uses four nonterminal layers followed by the pivot terminal. These choices determine the number of independently sampled commitment matrices.

### 7.3 Statistical and computational parameters

For  $k = 1024$ , the statistical error bound is below  $2^{-130}$ , with  $-\log_2 \kappa_{\text{alg}} \approx 131.976$ . The honest probability that either 160-attempt retry loop is exhausted at any of the five layers is at most  $10 \cdot 2^{-160}$ . Appendix D.8 gives the error decomposition and the values for the other arities.

The large-arity profile has seventeen independently sampled commitment matrices: three front-end matrices, three matrices at each of four nonterminal layers, and two at the pivot terminal. The official lattice estimator [A+26] was run on every role in classical and quantum MATZOV cost models [MAT22]. Under the coefficient-expanded Euclidean SIS model, the smallest reported costs occur for  $A_1$ : 155.624 and 148.847 in  $\log_2(\text{rop})$ , both at BKZ block size 441. The complete role table and all extrapolation flags appear in Appendix D.9.

The estimates describe the cost of the modeled lattice attacks. They do not evaluate the full advantage functions or the extraction-time term in Theorem 3.

### 7.4 Communication and operation counts

A base-field element occupies six bytes, an extension-field element occupies 24 bytes, and a ring element occupies 384 bytes. Each aggregation challenge contains seven base-field elements, or 42 bytes. Projection and terminal responses use the

fixed-padded signed Rice codec specified in Appendix E. Every message is byte-aligned and every retry attempt carries a one-byte tag.

The  $k = 1024$  no-retry interaction consists of 120 224 prover bytes and 46 355 verifier bytes, for 166 579 bytes in total. Transporting the 1025 incoming state commitments would add 1 574 400 bytes. The CRS and application-level public-input envelope remain separate. Appendix E gives the directional ledger, retry maxima, and phase-by-phase syntax.

Beyond the carrier costs in Table 2, the field stage requires sparse matrix applications, sumcheck, and equality-table operations. The lattice stage adds ring products, projection additions, generator expansion, and coefficient operations. Appendix E counts these in their respective arithmetic units.

The compiler, parameters, and scripts for these calculations are available in the accompanying repository.<sup>1</sup> Communication is counted from the message syntax, and arithmetic from the specified algorithms; no wall-clock timings are reported.

## 8 Conclusion

The Cauchy representation of [WX26] removes the need to retain a separate mixed term for every source pair. CauchyFold makes this representation usable in a lattice protocol: its coefficients can be computed without enumerating the pairs, and the folded values can be committed with a bounded honest opening. For fixed relation dimensions, preparing the mixed contribution uses quasi-linear arithmetic in the arity.

Canonical re-encoding preserves the accumulator layout and honest opening bound. The finite-node analysis recovers source witnesses or a short commitment kernel, with explicit statistical loss and expected extraction time. The specialized compiler and message encodings give 162.67 KiB of no-retry bidirectional communication at  $k = 1024$ , excluding incoming commitments and the CRS.

*Use of large language models.* Large language models were used to polish the text, assist with code, and check mathematical rigor and grammar. They were not used to generate the main ideas or principal proofs of this paper. The author has reviewed all model-assisted text and code and takes full responsibility for the paper and its accompanying code.

## References

- A<sup>+</sup>26. M. R. Albrecht et al. Security estimates for lattice problems. Sage software, 2026.
- Ajt96. M. Ajtai. Generating hard instances of lattice problems (extended abstract). In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing (STOC 1996)*, pages 99–108. ACM, 1996.

---

<sup>1</sup> <https://github.com/cxcx52/cauchyfold>

- AKLY24. T. Attema, M. Klooß, R. W. F. Lai, and P. Yatsyna. Adaptive special soundness: A new framework for knowledge extraction with applications to Kilian’s protocol. *Cryptology ePrint Archive*, Paper 2024/2038, 2024.
- BC23. B. Bünz and B. Chen. Protostar: Generic efficient accumulation/folding for special-sound protocols. In *Advances in Cryptology—ASIACRYPT 2023, Part II, Lecture Notes in Computer Science* 14439, pages 77–110. Springer, 2023.
- BC25. D. Boneh and B. Chen. LatticeFold+: Faster, simpler, shorter lattice-based folding for succinct proof systems. In *Advances in Cryptology—CRYPTO 2025, Part VII, Lecture Notes in Computer Science* 16006, pages 327–361. Springer, 2025.
- BC26. D. Boneh and B. Chen. LatticeFold: A lattice-based folding scheme and its applications to succinct proof systems. In *Advances in Cryptology—ASIACRYPT 2025, Part III, Lecture Notes in Computer Science* 16247, pages 330–362. Springer, 2026.
- BCFW25. B. Bünz, A. Chiesa, G. Fenzi, and W. Wang. Linear-time accumulation schemes. *Cryptology ePrint Archive*, Paper 2025/753, 2025.
- BCMS20. B. Bünz, A. Chiesa, P. Mishra, and N. Spooner. Recursive proof composition from accumulation schemes. In *Theory of Cryptography—TCC 2020, Part II, Lecture Notes in Computer Science* 12551, pages 1–18. Springer, 2020.
- BCS21. J. Bootle, A. Chiesa, and K. Sotiraki. Sumcheck arguments and their applications. *Cryptology ePrint Archive*, Paper 2021/333, 2021.
- BDL<sup>+</sup>16. C. Baum, I. Damgård, V. Lyubashevsky, S. Oechsner, and C. Peikert. More efficient commitments from structured lattice assumptions. *Cryptology ePrint Archive*, Paper 2016/997, 2016.
- BGH19. S. Bowe, J. Grigg, and D. Hopwood. Recursive proof composition without a trusted setup. *Cryptology ePrint Archive*, Paper 2019/1021, 2019.
- BLS03. A. Bostan, G. Lecerf, and É. Schost. Tellegen’s principle into practice. In *Proceedings of ISSAC 2003*, pages 37–44. ACM, 2003.
- BMNW24. B. Bünz, P. Mishra, W. Nguyen, and W. Wang. Arc: Accumulation for reed–solomon codes. *Cryptology ePrint Archive*, Paper 2024/1731, 2024.
- BNP26. D. Balbás, A. Nitulescu, and M. Plançon. ProtogaLattice: Lattice-based algebraic folding. *Cryptology ePrint Archive*, Paper 2026/1317, 2026.
- BS05. A. Bostan and É. Schost. Polynomial evaluation and interpolation on special sets of points. *Journal of Complexity*, 21(4):420–446, 2005.
- Che25. B. Chen. Symphony: Scalable SNARKs in the random oracle model from lattice-based high-arity folding. *Cryptology ePrint Archive*, Paper 2025/1905, 2025.
- DP25. B. E. Diamond and J. Posen. Succinct arguments over towers of binary fields. In *Advances in Cryptology—EUROCRYPT 2025, Part IV, Lecture Notes in Computer Science* 15604, pages 93–122. Springer, 2025.
- GLLO26. A. Garreta, H. Lipmaa, U. Luhaäär, and M. Osadnik. Cyclo: Lightweight lattice-based folding via partial range checks. In *Advances in Cryptology—EUROCRYPT 2026, Part VII, Lecture Notes in Computer Science* 16547, pages 33–62. Springer, 2026.
- GLS<sup>+</sup>21. A. Golovnev, J. Lee, S. Setty, J. Thaler, and R. S. Wahby. Brakedown: Linear-time and field-agnostic SNARKs for R1CS. *Cryptology ePrint Archive*, Paper 2021/1043, 2021.

- KLN<sup>+</sup>26. M. Klooss, R. W. F. Lai, N. K. Nguyen, M. Osadnik, and L. Tucci. RoKoko: Lattice-based succinct arguments, a committed refinement. Cryptology ePrint Archive, Paper 2026/575, 2026.
- KLNO24. M. Klooß, R. W. F. Lai, N. K. Nguyen, and M. Osadnik. RoK, paper, SISsors – toolkit for lattice-based succinct arguments. Cryptology ePrint Archive, Paper 2024/1972, 2024.
- KLNO25. M. Klooß, R. W. F. Lai, N. K. Nguyen, and M. Osadnik. RoK and roll – verifier-efficient random projection for  $\tilde{O}(\lambda)$ -size lattice arguments. Cryptology ePrint Archive, Paper 2025/1220, 2025.
- KLOT25. S. Kuriyama, R. W. F. Lai, M. Osadnik, and L. Tucci. SALSAA – sumcheck-aided lattice-based succinct arguments and applications. Cryptology ePrint Archive, Paper 2025/2124, 2025.
- KP23. A. Kothapalli and B. Parno. Algebraic reductions of knowledge. In *Advances in Cryptology—CRYPTO 2023, Part IV, Lecture Notes in Computer Science* 14084, pages 669–701. Springer, 2023.
- KS24. A. Kothapalli and S. Setty. HyperNova: Recursive arguments for customizable constraint systems. In *Advances in Cryptology—CRYPTO 2024, Part X, Lecture Notes in Computer Science* 14929, pages 345–379. Springer, 2024.
- KST22. A. Kothapalli, S. Setty, and I. Tzialla. Nova: Recursive zero-knowledge arguments from folding schemes. In *Advances in Cryptology—CRYPTO 2022, Part IV, Lecture Notes in Computer Science* 13510, pages 359–388. Springer, 2022.
- LFKN92. C. Lund, L. Fortnow, H. Karloff, and N. Nisan. Algebraic methods for interactive proof systems. *Journal of the ACM*, 39(4):859–868, 1992.
- LS15. A. Langlois and D. Stehlé. Worst-case to average-case reductions for module lattices. *Designs, Codes and Cryptography*, 75(3):565–599, 2015.
- LS18. V. Lyubashevsky and G. Seiler. Short, invertible elements in partially splitting cyclotomic rings and applications to lattice-based zero-knowledge proofs. In *Advances in Cryptology—EUROCRYPT 2018, Part I, Lecture Notes in Computer Science* 10820, pages 204–224. Springer, 2018.
- MAT22. MATZOV. Report on the security of LWE: Improved dual lattice attack. Technical report, The Center of Encryption and Information Security, IDF, 2022.
- Nis92. N. Nisan. Pseudorandom generators for space-bounded computation. *Combinatorica*, 12(4):449–461, 1992.
- NS26. W. Nguyen and S. Setty. Neo and SuperNeo: Post-quantum folding with pay-per-bit costs over small fields. In *Advances in Cryptology—CRYPTO 2026, Part IX, Lecture Notes in Computer Science* 16808, pages 492–523. Springer, 2026.
- Osa26. M. Osadnik. PikkuFold: Efficient folding in a few kilobytes. Cryptology ePrint Archive, Paper 2026/1809, 2026.
- Sch80. J. T. Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *Journal of the ACM*, 27(4):701–717, 1980.
- Set19. S. Setty. Spartan: Efficient and general-purpose zkSNARKs without trusted setup. Cryptology ePrint Archive, Paper 2019/550, 2019.
- vzGG13. J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, 3 edition, 2013.
- WF26. X. Wang and S. Fu. Compare before clearing: Exact integral-comparison frontiers for lattice extraction. Cryptology ePrint Archive, Paper 2026/1962, 2026.

- WX26. X. Wang and J. Xiang. The residue is the resource: Exact fresh-state complexity, reconstruction limits, and challenge design in folding. Cryptology ePrint Archive, Paper 2026/2019, 2026.
- XZZ<sup>+</sup>19. T. Xie, J. Zhang, Y. Zhang, C. Papamanthou, and D. Song. Libra: Succinct zero-knowledge proofs with optimal prover computation. Cryptology ePrint Archive, Paper 2019/317, 2019.

## A Carrier Algebra and Algorithms

This appendix records the polynomial routines used in Section 4. The routines depend only on the public poles, scales, basis maps, and source vectors, so they run before the verifier’s Cauchy challenge.

### A.1 Numerator and evaluation trees

For an index set  $I \subseteq \{1, \dots, k\}$ , write

$$D_I(T) = \prod_{i \in I} (T - \xi_i), \quad N_I(T) = \sum_{i \in I} a_i \frac{D_I(T)}{T - \xi_i},$$

where the  $a_i$  lie in any fixed finite-dimensional  $\mathbb{K}$ -vector space. The leaves contain  $D_{\{i\}} = T - \xi_i$  and  $N_{\{i\}} = a_i$ . For disjoint children  $I_0, I_1$ , combine them by

$$D_I = D_{I_0} D_{I_1}, \quad N_I = N_{I_0} D_{I_1} + N_{I_1} D_{I_0}. \quad (39)$$

Each summand acquires precisely the missing denominator factors, so induction on the tree proves the displayed definition of  $N_I$ . For carrier construction, the weights are respectively  $\lambda_i z_i$ ,  $\lambda_i^2 Q(z_i)$ , or  $J(R_i)$ .

At a level whose subproblems have size at most  $m$ , the total scalar work is  $O((k/m)M(m)) = O(M(k))$ . The equality of these bounds follows from the assumptions on  $M$ . There are  $O(\log(k+1))$  levels. For an  $a$ -dimensional numerator, applying the recurrence to every coordinate costs  $O(aP(k))$ . Unequal subtrees handle values of  $k$  that are not powers of two.

For evaluation at distinct points  $t_1, \dots, t_k$ , construct the product tree of  $T - t_j$ . Starting at its root, reduce the input polynomial modulo each child’s product. The leaf remainders are the evaluations. This remainder tree has the same asymptotic cost as the numerator tree. It applies to each coordinate of  $N$  and to the derivatives used in the two algorithms. The scalar tree is shared by all coordinates.

Interpolation also uses (39). If  $L(T) = \prod_j (T - t_j)$  and the values are  $h_j$ , form

$$F(T) = \sum_j \frac{h_j}{L'(t_j)} \frac{L(T)}{T - t_j}.$$

Distinctness gives  $L'(t_j) \neq 0$  and  $F(t_j) = h_j$ . Since  $\deg F < k$ , this is the unique interpolant. The evaluations of  $L'$  and the weighted reconstruction give

an  $O(aM(k)\log(k+1))$  algorithm for  $a$ -dimensional data. These polynomial routines are standard [vzGG13,BS05]. Transposed multiplication and division also give fast evaluation and interpolation routines [BLS03]. Here the trees avoid explicit Cauchy matrix multiplication.

## A.2 Multipoint construction in detail

The rational function  $M_Q/D$  has simple poles:

$$\frac{M_Q(T)}{D(T)} = \sum_i \frac{\lambda_i^2 Q(z_i)}{T - \xi_i}.$$

Differentiate in  $\mathbb{K}(T)$ , coordinate by coordinate. The quotient rule gives

$$-\left(\frac{M_Q}{D}\right)' = \frac{M_Q D' - M_Q' D}{D^2} = \sum_i \frac{\lambda_i^2 Q(z_i)}{(T - \xi_i)^2}. \quad (40)$$

This uses formal derivatives and involves no division by factorials. The identity is valid in positive characteristic as well.

A complete execution proceeds as follows.

1. Construct  $D, N, M_Q$  by the numerator tree. If source values  $Q(z_i)$  are not part of the available data, evaluate them first.
2. Form  $D', M_Q'$  and evaluate  $D, D', N, M_Q, M_Q'$  at the fixed nonpole points  $\tau_1, \dots, \tau_k$ .
3. Invert the nonzero  $D(\tau_j)$  and compute the vectors  $s_j$  and  $h_j$  in Section 4.3. Equation (40) removes the diagonal source terms.
4. Interpolate the  $h_j$  in ambient  $Y$ . Apply  $J$  to each coefficient to obtain the serialized image-coordinate order.

The need for  $k$  distinct nonpole points is an algorithmic precondition of this version. At  $k = 16$ , the concrete construction uses  $\xi_i = i - 1$  and  $\tau_j = 15 + j$ , for  $1 \leq i, j \leq 16$ . They lie in  $\mathbb{F}_q \subset \mathbb{K}$  and are distinct because  $q > 31$ . They are not sampled during the protocol.

## A.3 Deriving the pole-residue formula

Fix a pole  $\xi_i$ . The rational source vector can be written locally as

$$z_0 + \frac{N(T)}{D(T)} = \frac{\lambda_i z_i}{T - \xi_i} + z_0 + \sum_{j \neq i} \frac{\lambda_j z_j}{T - \xi_j}.$$

The second part is regular at  $\xi_i$  and takes the value  $v_i$ . After the double-pole diagonal term in (5) is subtracted, the coefficient of  $(T - \xi_i)^{-1}$  is  $R_i = \lambda_i B(z_i, v_i)$ . The following elementary calculation also obtains it without manipulating Laurent series.

Since  $D = (T - \xi_i)P_i$ , formal differentiation gives

$$D'(\xi_i) = P_i(\xi_i), \quad D''(\xi_i) = 2P_i'(\xi_i).$$

For  $j \neq i$ , the polynomial  $P_j$  contains the factor  $T - \xi_i$ , so

$$P_j'(\xi_i) = \prod_{h \neq i, j} (\xi_i - \xi_h) = \frac{D'(\xi_i)}{\xi_i - \xi_j}.$$

Consequently

$$\begin{aligned} N'(\xi_i) &= \lambda_i z_i P_i'(\xi_i) + \sum_{j \neq i} \lambda_j z_j P_j'(\xi_i) \\ &= \lambda_i z_i D''(\xi_i)/2 + D'(\xi_i) \sum_{j \neq i} \frac{\lambda_j z_j}{\xi_i - \xi_j}. \end{aligned}$$

Dividing by  $D'(\xi_i)$  and subtracting the first summand yields the formula for  $v_i$  in (6). The only characteristic restriction is that 2 be invertible. In particular, no condition  $\text{char } \mathbb{K} > k$  is needed for this derivative argument.

The pairwise identity in the proof of Theorem 1 verifies reconstruction from these residues. Since  $R_i \in Y_B$ , we may apply  $J$  before the final numerator tree, reducing its output dimension from  $y$  to  $r$ .

For  $k = 1$ , the formula specializes to  $D'' = 0$ ,  $v_1 = z_0$ , and  $H = \lambda_1 B(z_1, z_0)$ . Thus the same algorithm covers this endpoint.

#### A.4 Inversions, basis changes, and the concrete count model

For nonzero  $a_1, \dots, a_k$ , all inverses can be obtained from the inverse of their product. Form the prefix products, invert the final product, and traverse the prefixes backwards to recover each  $a_i^{-1}$ . This uses one field inversion and  $O(k)$  multiplications. It applies both to  $D(\tau_j)$  and to  $D'(\xi_i)$ . Interpolation denominators can be handled in the same way or precomputed with the fixed interpolation nodes. The arithmetic bounds of Section 4 allow either implementation.

The operation count uses schoolbook polynomial arithmetic and direct inversions for fixed public preparation. The asymptotic bounds above also allow fast multiplication and batch inversion.

For the concrete relation at  $k = 16$ , one  $Q$  evaluation uses 36 multiplications and 32 additions or subtractions in  $\mathbb{K}$ . One direct  $B$  evaluation uses 72 multiplications and 68 additions or subtractions. The maps  $J, U$  are identities on  $\mathbb{K}^4$ , so each application copies four coordinates and uses no field arithmetic. With the source  $Q$ -values cached, the multipoint algorithm makes 16 new  $Q$  calls. Without that cache it makes 33 in total. The pole-residue algorithm makes 16 calls to  $B$  in either case. These map calls are expanded in the operation counts; they are not added a second time to the expanded multiplication and addition counts.

For a general relation, a dense  $J$  can dominate the coordinate conversion. Its cost is therefore explicit in (7) and (8). The coefficient representation alone does

not provide an efficient basis map. If the basis is computed from the values of  $B$  on source basis pairs, both those evaluations and the ensuing linear algebra belong to relation preprocessing.

## B Encoding and Compiler Specification

The deterministic compiler fixes the relation, public-coordinate map, committed bit segments, and field constraint matrices. This appendix specifies the common relation and the arity-dependent layouts used in Section 7.

### B.1 Semantic relation and mixed-output coordinates

The source space is  $S = \mathbb{K}^{69}$ , with coordinates ordered as

$$z = (u, x_0, x_1, x_2, x_3, (a_{j,l}, b_{j,l})_{j=0,\dots,3; l=0,\dots,7}).$$

Within the last group,  $j$  is the outer index and  $l$  the inner index; each  $a_{j,l}$  is followed by  $b_{j,l}$ . Define

$$Q_j(z) = \sum_{l=0}^7 a_{j,l} b_{j,l} - u x_j, \quad 0 \leq j < 4. \quad (41)$$

Thus  $Y = \mathbb{K}^4$ . The public-coordinate map selects  $u, x_0, x_1, x_2, x_3$ . A fresh record has  $u = 1$ , residual  $E = 0$ , and all its  $z$  coordinates in the embedded base field. Its four equations require the indicated dot products to equal the public values  $x_j$ .

An accumulator satisfies  $Q(z) = E$  over  $\mathbb{K}$ , with its public coordinates fixed by the instance. The public statement fixes the selected coordinates, while the remaining coordinates and the committed residual form its opening.

Let  $e_{a_{j,0}}, e_{b_{j,0}}$  denote the corresponding standard source basis vectors, and let  $e_j$  denote an output basis vector. Then  $B(e_{a_{j,0}}, e_{b_{j,0}}) = e_j$ . These four explicit pairs span  $Y$ , while  $B$  always takes values in  $Y$ . It follows that  $Y_B = Y$ ,  $r = y = 4$ , and  $J = U = I_4$ . This algebraic calculation fixes the basis used by the compiler.

### B.2 Canonical field values and comparison helpers

The field representation is

$$q = 281474976710597, \quad \mathbb{K} = \mathbb{F}_q[\theta]/(\theta^4 - 4\theta^2 + 2).$$

An element of  $\mathbb{K}$  is encoded in the basis  $(1, \theta, \theta^2, \theta^3)$ . Each of its four coefficients is the unique integer in  $[0, q)$  representing that base-field element. Its binary

encoding has 48 bits, least significant first. Therefore one  $\mathbb{K}$  value occupies 192 coefficient bits. Decoding is the linear map

$$\text{Dec}(b) = \sum_{v=0}^3 \theta^v \sum_{j=0}^{47} 2^j b_{v,j} \quad \text{in } \mathbb{K}.$$

Range constraints make this linear decoding canonical. Boolean constraints alone would also accept the 48-bit representation of  $q$ .

For one base-field coefficient, let  $q_j$  be the binary digits of  $q$ . Store Boolean helpers  $e_0, \dots, e_{47}$ , and let  $e_{48}$  alias the fixed-one coordinate. For  $j = 47, \dots, 0$ , enforce

$$\begin{aligned} e_j &= \begin{cases} e_{j+1}b_j, & q_j = 1, \\ e_{j+1}(1 - b_j), & q_j = 0, \end{cases} \\ e_{j+1}b_j &= 0 \text{ if } q_j = 0, \quad e_0 = 0. \end{aligned} \tag{42}$$

Every value bit and every helper is constrained to be Boolean.

**Lemma 1 (Canonical comparison).** *For Boolean digits  $b_0, \dots, b_{47}$ , the constraints in (42) have a solution in the helper bits exactly when  $\sum_{j=0}^{47} 2^j b_j < q$ . The solution, when it exists, is unique.*

*Proof.* The recurrence forces  $e_j = 1$  precisely when all digits from position 47 down to  $j$  equal those of  $q$ . Starting from  $e_{48} = 1$  determines every helper. At the first differing digit, a zero digit of  $q$  cannot be replaced by a one, because  $e_{j+1}b_j = 0$  would fail. A one digit of  $q$  may be replaced by zero; the equality flag then becomes zero and remains zero. Finally,  $e_0 = 0$  excludes equality with  $q$ . This is exactly strict comparison of the represented nonnegative integers. The forced helpers are Boolean, so they satisfy the additional helper Boolean constraints.

**Lemma 2 (Weight of a canonical coefficient).** *Let  $x \in [0, q)$ , and include both its 48 value bits and the 48 comparison helpers from (42). Their joint Hamming weight is at most 90. This bound is attained by  $x = q - 1$ .*

*Proof.* Let  $t$  be the highest position at which the binary expansions of  $x$  and  $q$  differ. Since  $x < q$ , we have  $q_t = 1$  and  $x_t = 0$ . The helper string then has  $47 - t$  ones. The value string has at most  $\text{wt}(q_{47} \cdots q_{t+1}) + t$  ones, since the  $t$  lower positions are unrestricted. Hence their joint weight is at most  $47 + \text{wt}(q_{47} \cdots q_{t+1}) \leq 90$ , because  $q$  has binary weight 44. For  $x = q - 1$ , the value string has weight 43 and the helper string has weight 47.

Each helper is one base-field bit, subject only to its Boolean and comparison constraints.

*Specialized comparison for the large-arity profile.* The identity  $q = 64(2^{42} - 1) + 5$  gives a shorter comparison. Let  $b_0, \dots, b_{47}$  be Boolean value bits. Introduce six Boolean bits  $h_0, \dots, h_5$  and enforce

$$\sum_{j=0}^5 2^j h_j = 42 - \sum_{j=6}^{47} b_j.$$

The right side counts zero bits in the high part. Define five product helpers by  $p_1 = (1 - h_0)(1 - h_1)$  and  $p_j = p_{j-1}(1 - h_j)$  for  $2 \leq j \leq 5$ . Put  $e = p_5$ , introduce  $m = eb_2$ , and require

$$e(b_3 + b_4 + b_5) = 0, \quad m(b_1 + b_0) = 0.$$

The count equation is an integer equality because both sides lie in  $[0, 63]$ , below  $q$ . It uniquely determines the count bits, and the product equations uniquely determine the remaining helpers. If the high part is not all ones, the represented value is below  $q$ . Otherwise  $e = 1$ , and the two restrictions allow exactly the low values  $0, \dots, 4$ . Thus the comparison accepts exactly the canonical values and has a unique helper extension. The product helpers are Boolean consequences of their defining equations. There are twelve helpers and 63 constraints per free coefficient: 48 value-bit and six count-bit Boolean constraints, one count equation, six product equations, and two low-part restrictions.

The joint weight of the value and helper bits is at most 49. If the high part has no zero bits, its weight is 42, the five product helpers are one, and the low part together with  $m$  has weight at most two. Otherwise let  $s > 0$  be its zero count. The count bits and the initial zero-test products have total weight at most  $s$ , while  $e = m = 0$ . Their contribution cancels the missing high bits, giving total weight at most 48. The bound 49 is attained at  $q - 1$ .

### B.3 Segments, packing, and field constraints

A state segment contains  $z$  followed by its four residual coordinates. Each state therefore occupies  $L_{\text{st}} = 192(69 + 4) = 14\,016$  bits, or 219 ring columns. The carrier contains the  $k$  coefficients of  $\hat{H}$ , with four output coordinates inside each coefficient. Its length is  $768k$  bits, or  $12k$  ring columns.

The compiler inlines affine combinations in the R1CS linear forms. Its private arithmetic gates are the 32 products  $a_{j,\ell}b_{j,\ell}$  and the four products  $ux_j$  needed to check  $Q(z^*) = E^*$ . Thus  $g_K = 36$  canonical  $\mathbb{K}$ -valued gates are added. The accumulator,  $k$  fresh records, folded output, carrier, and gate values contain

$$V_K = 73(k + 2) + 4k + 36 = 77k + 182$$

canonical extension-field values. In the reference encoding, their comparison helpers occupy  $192V_K$  bits. Including the gate encodings, the auxiliary segment has

$$L_{\text{aux}} = 192g_K + 192V_K = 14\,784k + 41\,856$$

bits, or  $231k + 654$  ring columns.

Packing places each consecutive group of 64 bits in the coefficients of an element of  $R_q$ , in increasing powers of  $X$ . The state matrix is reused by the accumulator, all fresh records, and the folded output. The carrier and auxiliary matrices are sampled independently. These logical commitments therefore use three front-end matrices.

For the reference encoding, the handoff order is the accumulator, the  $k$  fresh records, the folded output, the carrier, the auxiliary segment, and the fixed-one coordinate. Its compact length is

$$N_{\text{used}}(k) = 29\,568k + 69\,889. \quad (43)$$

Zeros introduced by a backend block partition receive explicit zero equations.

Let  $\mathbf{b} \in \mathbb{F}_q^{N_{\text{used}}(k)}$  be this handoff vector. Its honest squared norm is smaller than its length. There are  $4V_K = 308k + 728$  encoded base-field coefficients. In each strict fresh record, the base-field condition on  $z_i \in \mathbb{K}^{69}$  fixes  $3 \cdot 69$  coefficients to zero,  $E_i = 0 \in \mathbb{K}^4$  fixes another 16, and  $u_i = 1$  fixes one coefficient to one. Thus  $223k$  coefficient encodings have zero weight,  $k$  have weight one, and  $84k + 728$  remain unrestricted. Lemma 2, together with the global fixed-one coordinate, gives

$$\|\mathbf{b}\|_2^2 \leq 90(84k + 728) + k + 1 = 7\,561k + 65\,521. \quad (44)$$

Every decoded semantic coordinate is a  $\mathbb{K}$ -linear form in  $\mathbf{b}$ . A multiplication gate is one row of

$$(\mathcal{A}\mathbf{b}) \odot (\mathcal{B}\mathbf{b}) - \mathcal{C}\mathbf{b} = 0 \quad \text{over } \mathbb{K}.$$

An affine equality is a row whose first factor is the fixed-one coordinate. The matrices check canonical encodings; public coordinates; strict and base-field fresh conditions; the two Cauchy fold equations; and  $Q(z^*) = E^*$ . Source validity and carrier consistency follow from the Cauchy comparison in Section 6.

For the reference encoding, the unpadded field-row count is

$$M_F(k) = 46\,109k + 108\,595. \quad (45)$$

The field-table index takes its first variable as the most significant bit. Padding adds only zero rows until the next power of two. The resulting depth is  $\ell_F = \lceil \log_2 M_F(k) \rceil$ .

Each of the three terminal  $\mathbb{K}$ -linear claims becomes four  $\mathbb{F}_q$ -linear claims because the handoff coordinates lie in the base field. Together with the commitment equations, the fixed-one equation, and any backend alignment zeros, these claims form the initial scalar system  $L_0 b = y_0$ .

#### B.4 Challenge independence and sparse-matrix representation

For  $c \notin \{0, \dots, k-1\}$ , the values  $(c-i)^{-1}$  and  $D(c)^{-1}$  are public coefficients. They change the coefficients of the fold equations and of the output public-coordinate equations. They never change a segment, gate, helper, row, or column allocation.

**Table 4.** Field constraints for the reference and specialized encodings.

| Constraint category                  | $k = 16$ | $k = 1024$ |
|--------------------------------------|----------|------------|
| Value-bit Boolean constraints        | 271 488  | 4 163 712  |
| Helper Boolean constraints           | 271 488  | 520 464    |
| Comparison recurrences and products  | 271 488  | 520 464    |
| High-part count equations            | 0        | 86 744     |
| Comparison restrictions              | 28 280   | 173 488    |
| Public coordinates                   | 90       | 4 106      |
| Explicit fresh-format equations      | 3 392    | 0          |
| Folded source and residual equations | 73       | 73         |
| Quadratic gates and output equations | 40       | 40         |
| Total field constraints              | 846 339  | 5 469 091  |
| Padded field rows                    | $2^{20}$ | $2^{23}$   |

The  $k = 16$  column uses the reference prefix comparator;  $k = 1024$  uses the specialized comparator and substitutes fixed fresh coordinates. Backend alignment zeros are excluded.

The compiler therefore emits one fixed matrix layout together with coefficient expressions parameterized by  $c$  and the public statement.

The matrices are stored in compressed sparse row form. Their coefficient identifiers refer to  $\mathbb{K}$  values and to public expressions in the challenge. For the  $k = 1024$  profile,  $\mathcal{A}, \mathcal{B}, \mathcal{C}$  contain 5 562 711, 19 555 250, and 527 376 stored entries, respectively. The allocated positions are fixed over the supported challenge set even when a public coefficient vanishes at one challenge. The sparse layouts determine the matrix-application and transpose counts used in Appendix E.

## C Finite Extraction and Statistical Proofs

### C.1 Comparing independent recovery trials

The comparison in Section 6.3 takes a recovery algorithm  $\mathcal{O}$  at one complete checkpoint. Independent calls restore the same public and prover state and use fresh continuation coins. A call returns failure, a verified short kernel, or a candidate. Write  $s = \Pr[\mathcal{O} \neq \perp]$ , and let  $\mu$  bound its entire expected cost.

Two incompatible candidates must yield a verifiable kernel by the specified comparison procedure. For each fixed bad candidate  $u$ , assume

$$\Pr[\mathcal{O} \text{ returns a candidate compatible with } u] \leq \varepsilon. \quad (46)$$

Compatibility depends on the objects fixed at this checkpoint, not on messages that may change in a fresh continuation.

**Lemma 3 (Independent-trial comparison).** *There is an algorithm that returns a good candidate or a kernel with probability at least  $s - \varepsilon$ . Its expected raw-trial cost is at most  $2\mu$ , in addition to comparison and checking.*

*Proof.* Run  $\mathcal{O}$  once. On failure, stop; on a kernel, return it. Otherwise retain the candidate  $u$  and repeat independent calls to  $\mathcal{O}$  until a nonfailure result is obtained. A kernel is returned immediately. Incompatible candidates produce their checked kernel. For compatible candidates, check the target predicate on  $u$  and return it only if it is good.

If  $s = 0$ , the loop is never reached. If  $s > 0$ , its last sample has the success-conditioned distribution of  $\mathcal{O}$ . For a fixed bad  $u$ , equation (46) bounds the probability of a compatible candidate in that distribution by  $\varepsilon/s$ . The total probability that the first call returns a bad candidate is at most  $s$ . Thus the additional failure probability is at most  $\varepsilon$ .

For cost, let  $C_n$  be the cost of trial  $n$  in an independent sequence. The event that the first  $n - 1$  calls fail is independent of  $C_n$ , even if  $C_n$  is correlated with its own outcome. Tonelli's theorem gives the expected cost until first success as

$$\sum_{n \geq 1} (1 - s)^{n-1} \mathbb{E} C_n \leq \frac{\mu}{s}.$$

The loop is reached with probability at most  $s$ , so it contributes at most  $\mu$  on average. The initial call contributes another  $\mu$ .

## C.2 Coordinate sampling with challenge-dependent success

Let the challenge space be  $\prod_{j=0}^{s-1} \mathcal{D}_j$  with  $|\mathcal{D}_j| = M_j \geq h$  for an integer  $h \geq 3$ . A call at challenge  $c$  succeeds with an arbitrary probability  $a(c)$ . Calls all restore the same complete short-challenge checkpoint. The success event can include a kernel, which terminates the algorithm immediately.

Choose a uniform center and call the oracle once. If successful, seek  $h - 1$  further successful values on every coordinate fiber. When  $r \in \{1, \dots, h - 1\}$  values of coordinate  $j$  are already excluded, draw uniformly from the remaining  $M_j - r$  values. On failure, stop with probability  $r/M_j$ ; otherwise repeat. A success supplies the next point. These stopping coins are extractor coins.

**Lemma 4 ( $h$ -point sampling).** *Let  $p$  be the success probability of a uniform center. The procedure returns a complete  $h$ -point star or a kernel with probability at least*

$$p - \binom{h}{2} \sum_{j=0}^{s-1} M_j^{-1}.$$

*It makes at most  $1 + (h - 1)s$  oracle calls in expectation.*

*Proof.* Fix all center coordinates except  $j$ , and let  $\delta$  be the average of  $a$  on the resulting fiber. If  $E$  is the excluded set of size  $r$ , the next attempt succeeds with probability

$$u = \frac{M_j \delta - \sum_{e \in E} a(e)}{M_j - r}.$$

The probability that an attempt either succeeds or stops is

$$D = u + (1 - u) \frac{r}{M_j} = \delta + \frac{r - \sum_{e \in E} a(e)}{M_j} \geq \delta.$$

Consequently the expected number of calls in this stage is  $1/D$ , and its stopping probability is  $(1 - u)r/(M_j D)$ . These formulas apply conditional on the history that selected  $E$ .

For each possible original center, the probability of reaching the stage is at most its original success probability  $a(c)$ . Averaging this bound over the original uniform center gives  $\delta$  on the fixed fiber. Since  $D \geq \delta$ , the stage costs at most one call in expectation and stops with mass at most  $r/M_j$ . A fiber with  $\delta = 0$  cannot reach the stage. Summing over  $r = 1, \dots, h-1$  and all coordinates proves both bounds.

This calculation is taken under the original center distribution. Conditioning first on reaching a later stage would generally bias that distribution. Uniform sampling with at most  $h-1$  excluded values can be implemented by rejection sampling, with expected overhead at most  $M_j/(M_j - h + 1) \leq h$  per retained draw.

For the actual bounded retry protocol, apply the lemma separately at each reached response attempt. Its checkpoint is fixed by the past; its current challenge is fresh. A retry in a coordinate fork counts as an unsuccessful oracle call. Summing over at most  $R_C$  attempts gives  $\binom{h}{2} R_C \sum_j M_j^{-1}$ . We use  $h = 5$ , giving loss  $10 R_C \sum_j M_j^{-1}$  and expected oracle count  $1 + 4s$ .

### C.3 Algebraic recovery and short integer comparisons

Suppose two accepted responses differ only in coordinate  $j$ . Their difference satisfies  $AX = t_j \Delta$ , where  $X$  is the difference of responses and  $\Delta$  the difference of challenges. The latter is a unit. Thus  $w_j = X \Delta^{-1}$  is a well-defined modular slope. Given two such certificates  $(X_1, \Delta_1)$  and  $(X_2, \Delta_2)$ , unequal slopes produce  $v = X_1 \Delta_2 - X_2 \Delta_1$ . Then  $Av = 0 \pmod q$ , and

$$v \pmod q = (w_j^{(1)} - w_j^{(2)}) \Delta_1 \Delta_2 \neq 0.$$

If every response has norm at most  $Z$  and every challenge has operator norm at most  $T^{\text{op}}$ , the shared center gives a sharper comparison than bounding the two products separately. Write the center response and challenge as  $(z, c)$  and the two off-center pairs as  $(z_1, c_1)$  and  $(z_2, c_2)$ . Commutativity gives

$$\begin{aligned} v &= (z_1 - z)(c_2 - c) - (z_2 - z)(c_1 - c) \\ &= c_2(z_1 - z) + c(z_2 - z_1) + c_1(z - z_2). \end{aligned}$$

The three responses lie in the radius- $Z$  coefficient ball. Hence

$$\begin{aligned} &\|z_1 - z\|_2^2 + \|z_2 - z_1\|_2^2 + \|z - z_2\|_2^2 \\ &= 3(\|z\|_2^2 + \|z_1\|_2^2 + \|z_2\|_2^2) - \|z + z_1 + z_2\|_2^2 \leq 9Z^2. \end{aligned}$$

Cauchy–Schwarz bounds the corresponding perimeter by  $3\sqrt{3}Z$ . Applying the multiplication-operator bound to the preceding expansion therefore yields

$$\|v\|_2 \leq 3\sqrt{3}T^{\text{op}}Z. \quad (47)$$

If both slopes agree, put  $w_j = w_j^{(1)} = w_j^{(2)}$ . For any aggregation  $e$ , define

$$P_{e,j}(T) = \left\langle \sum_a \mathbf{c}_a(T) \phi_a^{(e)}, z + (T - \mathbf{c}_j)w_j \right\rangle - \sum_{a,b} \mathbf{c}_a(T) \mathbf{c}_b(T) h_{ab}^{(e)},$$

where only coordinate  $j$  depends on  $T$ . This degree-two polynomial vanishes at the center and both new coordinate values. Its evaluation matrix has determinant equal, up to sign, to the product of their pairwise differences. That determinant is a unit, so the polynomial is zero over  $R_q$ . Its quadratic coefficient yields  $\langle \phi_j^{(e)}, w_j \rangle = h_{jj}^{(e)}$ . The first response equation gives  $Aw_j = t_j$ . These facts suffice to recover all aggregate scalar equations.

At a nonterminal layer, any discrepancy in the recovered  $\hat{t}$  or  $\hat{h}$  is handled before this polynomial argument. Both vectors open commitments fixed at the short-challenge checkpoint. Their differences have norm at most  $2\gamma_{i+1}$ . At the terminal,  $t_1, \dots, t_{s-1}$  and  $h$  are public and fixed. Two different reconstructions of  $t_0$  have different canonical pivot digit vectors. Their difference is nonzero modulo  $q$ , opens the same pivot commitment to zero, and has norm at most  $\rho_p \sqrt{da_L \ell_p}$ .

**Lemma 5 (Comparison across independent trials).** *Let  $h \geq 3$ . Suppose two recovery trials for the same block commitment produce pairs  $(c_a, z_a)$  and  $(\tilde{c}_a, \tilde{z}_a)$ ,  $0 \leq a < h$ . Within each trial, challenge differences are units, responses have coefficient norm at most  $Z$ , and challenge multiplication operators have norm at most  $T^{\text{op}}$ . Suppose every pair in the first trial has modular slope  $w$ , every pair in the second has slope  $\tilde{w}$ , and  $Aw = A\tilde{w} = t$ . If  $w \neq \tilde{w}$ , one can find a nonzero modular kernel of  $A$  with integer norm at most*

$$\frac{4h}{h-1}T^{\text{op}}Z. \quad (48)$$

For  $h = 5$ , this is  $5T^{\text{op}}Z$ .

*Proof.* For each edge  $e = (a, b)$  with  $a < b$ , set  $D_e = c_b - c_a$  and  $X_e = z_b - z_a$ , using the bounded integer representatives. Define  $\tilde{D}_f, \tilde{X}_f$  in the other trial. Commutativity and the equal commitment images give

$$K_{e,f} = \tilde{D}_f X_e - D_e \tilde{X}_f, \quad AK_{e,f} = 0 \pmod{q}.$$

Modulo  $q$ , this vector equals  $D_e \tilde{D}_f (w - \tilde{w})$ , which is nonzero because both differences are units. Thus all  $\binom{h}{2}$  candidates are nonzero modular kernels.

The Euclidean pair identity gives

$$\sum_e \|X_e\|_2^2 = h \sum_a \|z_a\|_2^2 - \left\| \sum_a z_a \right\|_2^2 \leq h^2 Z^2.$$

Applied to the vectors  $\tilde{c}_a x$ , the same identity gives, for every fixed integer ring vector  $x$ ,

$$\sum_f \|\tilde{D}_f x\|_2^2 \leq h \sum_a \|\tilde{c}_a x\|_2^2 \leq h^2 (T^{\text{op}})^2 \|x\|_2^2.$$

Consequently  $\sum_{e,f} \|\tilde{D}_f X_e\|_2^2 \leq h^4 (T^{\text{op}})^2 Z^2$ , and the same bound holds for the other product. Using  $\|u - v\|_2^2 \leq 2\|u\|_2^2 + 2\|v\|_2^2$  yields

$$\sum_{e,f} \|K_{e,f}\|_2^2 \leq 4h^4 (T^{\text{op}})^2 Z^2.$$

Divide by  $\binom{h}{2}^2$  and take square roots. Enumerating these candidates and choosing one of least squared norm realizes the bound using integer arithmetic.

The local recovery checks all four slopes against the same center. Their agreement implies the pairwise slope condition in the lemma by subtraction. Their disagreement is handled by (47). Independent trials therefore need no common center, and both branches fit the registered  $3\sqrt{3}T^{\text{op}}Z$  radius.

#### C.4 A modular projection bound

The projection certifies the centered modular slope, rather than the short numerator used to recover it. The following bound is uniform over every fixed candidate vector.

**Lemma 6 (One-row test).** *Let  $\beta \geq 1$ ,  $q \geq 128\beta$ , and let  $w$  be a centered integer coefficient vector with  $\|w\|_2 \geq \beta$ . Let the coordinates of  $\pi$  be independent, with  $\Pr[\pi_j = -1] = \Pr[\pi_j = 1] = 1/4$  and  $\Pr[\pi_j = 0] = 1/2$ . Then*

$$\mathbb{E}_\pi \exp\left(-\frac{\text{cen}(\langle \pi, w \rangle)^2}{3\beta^2}\right) \leq \frac{7}{8} + 2^{-1000}.$$

*Proof.* Consider the  $q$ -periodic Gaussian

$$G(x) = \sum_{l \in \mathbb{Z}} e^{-(x+lq)^2/(3\beta^2)}.$$

Its absolutely convergent Fourier expansion is

$$G(x) = \frac{\sqrt{3\pi}\beta}{q} \sum_{h \in \mathbb{Z}} e^{-3\pi^2\beta^2 h^2/q^2} e^{2\pi i h x/q}.$$

Write  $S_w = \sum_j \pi_j w_j$ . Its characteristic function at frequency  $h/q$  is

$$\mathbb{E} e^{2\pi i h S_w/q} = \prod_j \cos^2(\pi h w_j/q) \in [0, 1].$$

Removing coordinates can only increase this product. All Fourier coefficients are nonnegative. Hence for every subvector  $v$  of  $w$ ,

$$\mathbb{E}e^{-\text{cen}(S_w)^2/(3\beta^2)} \leq \mathbb{E}G(S_w) \leq \mathbb{E}G(S_v). \quad (49)$$

If some  $|w_j| \geq \beta$ , retain that coordinate alone. The  $l = 0$  contribution is at most  $(1 + e^{-1/3})/2 \leq 7/8$ . The coordinate is centered, so  $|w_j| \leq q/2$ . For  $l \neq 0$ , the distance to the  $l$ th Gaussian peak is at least  $|l|q/2$ , giving a total contribution at most  $2 \sum_{l \geq 1} e^{-l^2(q/\beta)^2/12} < 2^{-1000}$ .

Otherwise every coordinate has magnitude below  $\beta$ . Retain coordinates until  $V = \sum_j v_j^2$  first reaches  $\beta^2$ . Then  $\beta^2 \leq V < 2\beta^2$  and

$$\mathbb{E}S_v^2 = V/2, \quad \mathbb{E}S_v^4 = \frac{3}{4}V^2 - \frac{1}{4} \sum_j v_j^4 \leq \frac{3}{4}V^2.$$

Using  $e^{-x} \leq 1 - x + x^2/2$  for  $x \geq 0$  bounds the zero peak by

$$1 - \frac{V}{6\beta^2} + \frac{V^2}{24\beta^4} \leq \frac{7}{8}.$$

For the other peaks, independence and  $\cosh(tv_j/2)^2 \leq e^{t^2 v_j^2/4}$  give  $\Pr[|S_v| \geq x] \leq 2e^{-x^2/V}$ . For a nonzero integer  $l$ , split according to whether  $|S_v| \geq |l|q/2$ . The former event has probability at most  $2e^{-l^2 q^2/(8\beta^2)}$ ; on its complement the Gaussian term is at most  $e^{-l^2 q^2/(12\beta^2)}$ . Summing both signs of  $l$ , the nonzero peaks contribute at most

$$4 \sum_{l \geq 1} e^{-l^2(q/\beta)^2/8} + 2 \sum_{l \geq 1} e^{-l^2(q/\beta)^2/12} \leq \frac{4}{2^{2048} - 1} + \frac{2}{2^{1365} - 1} < 2^{-1000}.$$

Equation (49) completes the proof.

**Corollary 1 (Independent rows).** Put  $\theta_P = 7/8 + 2^{-1000}$ . For  $\gamma = \sigma\sqrt{S}$ ,  $\sigma \geq 3$ ,  $q \geq 128\gamma$ , and a fixed centered  $w$  with  $\|w\|_2 \geq \gamma$ , an  $m$ -row projection satisfies

$$\Pr[\|\text{cen}(Hw)\|_2^2 \leq mS] \leq \left(e^{1/(3\sigma^2)}\theta_P\right)^m \leq \left(\frac{3\sigma^2}{3\sigma^2 - 1}\theta_P\right)^m.$$

*Proof.* On the event in question,

$$\exp\left(-\frac{\|\text{cen}(Hw)\|_2^2}{3\gamma^2}\right) \geq e^{-m/(3\sigma^2)}.$$

Markov's inequality and row independence bound its probability by  $e^{m/(3\sigma^2)}\theta_P^m$ . The last inequality uses  $e^x \leq (1-x)^{-1}$  for  $0 \leq x < 1$ .

The last expression is rational whenever  $\sigma^2$  is rational, so its comparison with a target error can be made by exact integer arithmetic. This moment estimate remains available for any admissible profile. The selected profiles use the sharper independent-row bound described below. The protocol may choose a projection based on earlier failed attempts. For a fixed bad candidate, each new matrix still satisfies the bound conditional on the past. Induction over the attempts bounds the chance that at least one of  $R_P$  trials passes by  $1 - (1 - \epsilon_P)^{R_P}$ . The simpler union bound  $R_P \epsilon_P$  remains valid.

### C.5 Transport to seeded projections

We now replace the  $R = 2mN$  independent input bits used by the lazy-ternary projection with the output of Nisan's generator [Nis92]. The seed includes the initial block and every hash description. All of these bits are freshly and independently sampled in each raw projection attempt.

**Lemma 7 (Fixed-test transport).** *Let  $F_w : \{0, 1\}^R \rightarrow \{0, 1\}$  be the complete projection PASS predicate for a vector  $w$  fixed before the seed. Suppose  $F_w$  is computed by a finite-state machine with at most  $2^{w_{\text{fsm}}}$  states. For block length  $b$ , depth  $h$ , and any  $\delta > 0$ , the recursive affine-Toeplitz generator satisfies*

$$|\Pr[F_w(G) = 1] - \Pr[F_w(U_R) = 1]| \leq (2^h - 1)\delta + \frac{2^{6w_{\text{fsm}}h}}{\delta^2 2^b}.$$

*Proof.* Nisan's good-hash lemma bounds the probability of a bad hash sequence by  $2^{6w_{\text{fsm}}h}/(\delta^2 2^b)$  [Nis92, Lemma 2]. Conditioned on good hashes, the recursive hybrid changes the acceptance probability of any Boolean event by at most  $(2^h - 1)\delta$ . Splitting on the good-hash event gives the stated inequality. The probability is over the initial block and every hash choice; keeping the hashes fixed and refreshing only the initial block would not satisfy this premise.

For completeness, an affine Toeplitz hash on  $b$  bits is

$$h_{a,t}(x)_j = t_j + \sum_{r=0}^{b-1} a_{r+j} x_r \pmod{2}, \quad |a| = 2b - 1, \quad |t| = b.$$

It requires  $3b - 1$  description bits. Starting from  $G_0(x) = x$ , define

$$G_h(x; h_1, \dots, h_h) = G_{h-1}(x; h_1, \dots, h_{h-1}) \parallel G_{h-1}(h_h(x); h_1, \dots, h_{h-1}).$$

The complete seed length is  $b + h(3b - 1)$ . The expansion may be truncated after  $R$  bits.

For an input of length  $N$  and energy threshold  $T = mS$ , the PASS computation consumes two bits per trit. Its state stores the current bit position, one pending bit, the residue modulo  $q$ , and the accumulated squared energy clipped at  $T + 1$ . It therefore has at most

$$(R + 1) 4q(T + 2)$$

states. The vector  $w$  is fixed in the transition function.

Put

$$w_{\text{fsm}} = \lceil \log_2((R+1)4q(T+2)) \rceil, \quad \varepsilon_{\text{seed}} = 2^{-150}.$$

For an integer block length  $1 \leq b < R$ , set

$$h = \left\lceil \log_2 \left\lceil \frac{R}{b} \right\rceil \right\rceil, \quad \delta = \frac{2\varepsilon_{\text{seed}}}{3(2^h - 1)}.$$

The first term in Lemma 7 is then  $2\varepsilon_{\text{seed}}/3$ . Requiring

$$2^b \geq \frac{27 \cdot 2^{6w_{\text{fsm}}} h (2^h - 1)^2}{4\varepsilon_{\text{seed}}^3} \quad (50)$$

bounds the second term by  $\varepsilon_{\text{seed}}/3$ . Thus the total transport error is at most  $\varepsilon_{\text{seed}}$ .

For each possible depth  $h$ , the parameter calculation selects the smallest integer  $b$  satisfying (50) and  $b2^h \geq R$ . It keeps the pair only if the preceding depth formula holds, then minimizes  $b + h(3b - 1)$ . Ties are resolved first by the smaller  $b$  and then by the smaller  $h$ . All comparisons use integers or exact rational numbers. Appendix D.7 records the resulting seed lengths.

The bound also applies to a vector selected from earlier adaptive history: condition on that history and sample the next complete seed independently. It compares acceptance probabilities of this fixed norm test, even though the seed is public.

For every selected profile, the independent-row input to the transport lemma is the ternary projection bound of PikkuFold [Osa26, Theorem 2], specialized to  $\lambda = 192$ ,  $m = 384$ ,  $\alpha = 41.21 = 4121/100$ , and modular-range parameter 176. We set  $\sigma_i^2 = 38400/4121$ , so  $\alpha\sigma_i^2 = 384$  and the acceptance threshold is exactly  $384S_i$ . The condition  $176\gamma_i \leq q$  is checked at every selected layer. The independent projection error is at most  $2^{-192}$ ; fixed-test transport therefore gives  $\epsilon_{P,i} = 2^{-192} + 2^{-150}$ .

## C.6 Structured aggregation

**Lemma 8 (Structured aggregation for a fixed residual).** *Let  $r \in \mathbb{F}_q^R \setminus \{0\}$*

*be fixed before  $(\zeta, \chi) \xleftarrow{\$} \mathbb{K} \times \mathcal{H}$ , and let the three rows be defined by (17). Put  $D = \lceil R/4 \rceil - 1$ . Then*

$$\Pr[\langle \alpha^{(e)}, r \rangle = 0 \text{ for } e = 1, 2, 3] \leq q^{-3} + (1 - q^{-3}) \min\{1, D/q^4\}.$$

*Proof.* Pad only the final group of four residual coordinates and define

$$P_r(T) = \sum_{b=0}^{\lceil R/4 \rceil - 1} \left( \sum_{s=0}^3 r_{4b+s} \theta^s \right) T^b.$$

Linear independence of the field basis makes  $P_r$  a nonzero polynomial of degree at most  $D$ . Hence  $a := \Pr[P_r(\zeta) = 0] \leq \min\{1, D/q^4\}$ . For a fixed nonzero

$z = P_r(\zeta)$ , the event  $\pi(\chi z) = 0$  asks that  $\chi$  lie in  $z^{-1} \ker \pi$ . This is a one-dimensional linear subspace of  $\mathbb{K}$ . Its intersection with the affine hyperplane  $\mathcal{H}$ , whose last basis coordinate is one, contains at most one point. Since  $|\mathcal{H}| = q^3$ , the conditional probability is at most  $q^{-3}$ . The three aggregated residuals equal  $\pi(\chi P_r(\zeta))$ , so the total probability is at most  $a + (1-a)q^{-3}$ .

The affine multiplier  $\chi$  is needed even for a one-block residual: without it,  $(0, 0, 0, 1)$  would always be hidden by projection onto the first three basis coordinates. The argument above instead bounds the joint test on any residual fixed before  $(\zeta, \chi)$ .

### C.7 Short-challenge moments and honest retries

For the concrete ring degree  $d = 64$ , the box challenge admits a sharper operator bound than its coefficient  $\ell_1$  norm. Let  $c(X) = \sum_{j=0}^{63} c_j X^j$  with  $|c_j| \leq 2$ . Negacyclic multiplication is unitarily diagonalized over the roots  $\omega_t = \exp((2t+1)\pi i/64)$  of  $X^{64} + 1$ , so

$$\|c\|_{\text{op}} = \max_t |c(\omega_t)|.$$

For each root,

$$\begin{aligned} |c(\omega_t)| &= \max_{\phi \in \mathbb{R}} \Re(e^{-i\phi} c(\omega_t)) \\ &\leq 2 \max_{\phi \in \mathbb{R}} \sum_{j=0}^{63} |\cos(\phi + j\pi/64)| = 2 \csc(\pi/128) < 82. \end{aligned}$$

Multiplication by the odd number  $2t+1$  permutes the angles modulo  $\pi$ . The last trigonometric identity follows by fixing the signs between consecutive zeros and summing the resulting geometric series. Thus  $T^{\text{op}} = 82$  is a valid integer operator bound for the full challenge box. Distinct challenges have coefficients differing by at most four. Under the field and cyclotomic conditions used here, the short-difference unit criterion [BC25] applies when  $4 < \sqrt{q/2}$ .

Each challenge coefficient has mean zero and variance two. For an integer ring vector  $\tilde{v}$ , expansion of each output coefficient and cancellation of cross terms gives

$$\mathbb{E} \|\tilde{c}\tilde{v}\|_2^2 = 2d \|\tilde{v}\|_2^2. \quad (51)$$

Centering modulo  $q$  can only reduce the absolute value of each coefficient. Thus for modular multiplication the conclusion is the inequality  $\mathbb{E} \|\text{cen}(cv)\|_2^2 \leq 2d \|v\|_2^2$ .

For several blocks, let  $Y_j = \text{cen}(c_j w_j)$ . Since  $q$  is odd and the challenge distribution is symmetric,  $\mathbb{E} Y_j = 0$ . The  $Y_j$  are independent, so

$$\mathbb{E} \left\| \text{cen} \left( \sum_j c_j w_j \right) \right\|_2^2 \leq \mathbb{E} \left\| \sum_j Y_j \right\|_2^2 = \sum_j \mathbb{E} \|Y_j\|_2^2 \leq \mu S.$$

The response energy is an integer. Consequently, if  $G + 1 \geq 2\mu S$ , then rejection implies an energy of at least  $G + 1$ , and Markov's inequality bounds its probability by  $1/2$ . At the pivot terminal, conditioning the first challenge on being nonzero preserves symmetry and raises its moment by at most  $M/(M - 1)$ . Equation (25) gives the same  $1/2$  success bound.

For the projection, the uncentered variance calculation is  $\mathbb{E}\|II \text{coef}(w)\|_2^2 = m\|w\|_2^2/2$ . Centering again decreases the norm. Hence an honest projection response under independent trits passes  $\|p\|_2^2 \leq mS$  with probability at least  $1/2 + 1/(2(mS + 1))$ , using the integrality of the energy in Markov's inequality. Direct substitution gives  $2^{-150} < 1/(2(mS + 1))$  in every layer. Lemma 7 therefore leaves seeded honest success at least  $1/2$ . Fresh challenges in each bounded retry family give exhaustion probability at most  $2^{-R_P}$  and  $2^{-R_C}$ , respectively. The response bound remains valid after projection succeeds, because the witness is fixed and the short challenges are newly sampled. A union bound across layers proves (30).

### C.8 Norm propagation through a child

The radix decomposition used by the honest prover is deterministic. For a non-negative coefficient  $x$ , repeatedly set  $x' = \lfloor (x + b_i^{\text{dig}} - 1)/\rho_i \rfloor$  and use the remainder  $x - \rho_i x'$ . The remainder lies in  $[-b_i^{\text{dig}} + 1, b_i^{\text{dig}}]$ . Apply this to  $|x|$  and restore the sign for negative  $x$ . After  $\ell_i = \lceil \log_{\rho_i} q \rceil$  digits, every centered coefficient has been represented. This fixes the tie convention at the radix endpoints.

The energy of this canonical representation is smaller than the coordinatewise bound  $\ell_i (b_i^{\text{dig}})^2$ . For an even radix  $\rho = 2b$ , let  $f_{\rho,\ell}(x)$  be the sum of the first  $\ell$  squared digits produced from a nonnegative integer  $x$ , and put

$$\mathcal{F}_{\rho,\ell}(B) = \max_{0 \leq x \leq B} f_{\rho,\ell}(x).$$

Set  $J = \lfloor (B + b - 1)/\rho \rfloor$ . For  $\ell \geq 1$ , the exact recurrence is  $\mathcal{F}_{\rho,\ell}(B) = B^2$  when  $J = 0$ , and otherwise

$$\mathcal{F}_{\rho,\ell}(B) = \max \left\{ b^2 + \mathcal{F}_{\rho,\ell-1}(J - 1), f_{\rho,\ell-1}(J) + \max\{(b - 1)^2, (B - \rho J)^2\} \right\}. \quad (52)$$

Indeed, quotient fibers  $0, \dots, J - 1$  are complete. Their low digit ranges over  $[-b + 1, b]$ , so its largest square is  $b^2$ ; the remaining digits encode the quotient. The final fiber has quotient  $J$  and low digit in  $[-b + 1, B - \rho J]$ , where convexity places the maximum at an endpoint. Negating all digits handles negative centered integers and does not change their energy. For  $q = 2^{48} - 59$  and  $B = (q - 1)/2$ , the recurrence gives

$$\mathcal{F}_{8,16}(B) = 249, \quad \mathcal{F}_{64,8}(B) = 8129, \quad \mathcal{F}_{128,7}(B) = 25537.$$

These values apply to the canonical encodings of  $\hat{t}$  and  $\hat{h}$ . The independent low-part bound for the raw split of  $z$  remains  $dn_i(b_i^{\text{dig}})^2$ .

For an honest split  $z = z_{\text{lo}} + \rho_i z_{\text{hi}}$ , there are  $dn_i$  low coefficients, each bounded by  $b_i^{\text{dig}}$ . The triangle inequality yields

$$\|z_{\text{hi}}\|_2 \leq \frac{\sqrt{G_i} + b_i^{\text{dig}} \sqrt{dn_i}}{\rho_i}.$$

The two committed digit segments have squared-energy bounds

$$E_{t,i} = da_i s_i \mathcal{F}_{\rho_i, \ell_i}((q-1)/2), \quad E_{h,i} = d\tau_i \frac{s_i(s_i+1)}{2} \mathcal{F}_{\rho_i, \ell_i}((q-1)/2).$$

Adding these to the bounds for  $z_{\text{lo}}$  and  $z_{\text{hi}}$  gives the sufficient child-energy condition

$$S_{i+1} \geq dn_i (b_i^{\text{dig}})^2 + E_{t,i} + E_{h,i} + \left\lceil \frac{G_i + dn_i (b_i^{\text{dig}})^2 + 2b_i^{\text{dig}} \sqrt{dn_i G_i}}{\rho_i^2} \right\rceil. \quad (53)$$

Its ceiling can be evaluated without floating-point arithmetic: after isolating the single square-root term, comparison with a candidate integer is decided by squaring nonnegative integers.

An extracted child only has the total norm bound  $\|x\|_2 \leq \gamma_{i+1}$ . Its blocks need not be canonical radix digits. Nevertheless, Cauchy-Schwarz gives

$$\|\text{cen}(z_{\text{lo}} + \rho_i z_{\text{hi}})\|_2 \leq \sqrt{1 + \rho_i^2} \sqrt{\|z_{\text{lo}}\|_2^2 + \|z_{\text{hi}}\|_2^2} \leq \sqrt{1 + \rho_i^2} \gamma_{i+1}.$$

Each auxiliary block also has norm at most  $\gamma_{i+1}$ . Combining these inequalities with the integer comparison above proves (28).

### C.9 Conditional costs and finite composition

The expected-time bound records actual work, including full-seed sampling and parsing, streaming projection expansion, state restoration, arithmetic, and verification. Let  $H_i$  uniformly bound the conditional cost of preparing and running one layer- $i$  continuation, excluding the child extractor. Let  $W_i^{(5)}$  cover bounded projection and aggregation preparation, modular slope recovery, shared-center comparisons, and final witness checks. It also includes constructing and comparing at most  $100s_i$  cross-trial kernels: each five-point fiber has ten edges in each trial. Each candidate requires two bounded-coefficient ring-vector products, subtraction, and an exact squared-norm comparison. A sufficient recurrence is

$$T_i^{\text{ext}} \leq W_i^{(5)} + 2R_{C,i}(1 + 4s_i)(H_i + T_{i+1}^{\text{ext}}), \quad i < L. \quad (54)$$

At  $i = L$ , direct terminal verification replaces the child cost, giving

$$T_L^{\text{ext}} \leq W_L^{(5)} + 2R_{C,L}(1 + 4s_L)H_L.$$

The field and Cauchy extraction steps satisfy

$$T_F^{\text{ext}} \leq 2(H_F + T_0^{\text{ext}}) + W_F^{\text{ext}}, \quad T_{\text{node}}^{\text{ext}} \leq 2(H_C + T_F^{\text{ext}}) + W_C^{\text{ext}}. \quad (55)$$

Here  $H_F, H_C$  include their prover continuations, and  $W_F^{\text{ext}}, W_C^{\text{ext}}$  include the comparisons and semantic checks.

The number of invocations is random and can depend on previous successes. Let  $N$  be this count, and let  $\mathcal{F}_{n-1}$  contain the complete history before call  $n$ . The event  $N \geq n$  is  $\mathcal{F}_{n-1}$ -measurable. If every reached checkpoint satisfies  $\mathbb{E}[C_n \mid \mathcal{F}_{n-1}] \leq H + T$ , then

$$\begin{aligned} \mathbb{E} \sum_{n=1}^N C_n &= \sum_{n \geq 1} \mathbb{E} [\mathbf{1}_{\{N \geq n\}} \mathbb{E}[C_n \mid \mathcal{F}_{n-1}]] \\ &\leq (H + T) \sum_{n \geq 1} \Pr[N \geq n] = (H + T) \mathbb{E}N. \end{aligned}$$

Tonelli applies because the costs are nonnegative. This argument includes histories that select unusual challenges or child checkpoints; it requires a uniform conditional cost bound there, rather than a bound averaged only over an honest challenge.

For success, let the actual forward execution select child checkpoint  $t$  with subprobability measure  $\nu$ . It selects at most one child, so  $\nu$  has mass at most one. If the child extractor succeeds with probability  $p_t - A_{i+1}$  or better at every  $t$ , averaging gives

$$\int (p_t - A_{i+1}) d\nu(t) \geq \int p_t d\nu(t) - A_{i+1}.$$

The coordinate sampler then applies to this raw oracle's actual success function. Its loss is additive; the child loss is not charged again at every requested fork. The independent-trial comparison doubles the expected trial cost and adds only the fixed-candidate return error. These two facts give (54) and the induction  $p_i - A_i$ .

Throughout this induction a successful child return includes its actual accepting center branch. Prefix it by the parent's recorded messages to retain one accepting continuation. It is not necessary that the extracted witness, when used by an honest prover, reproduce all those messages. The record serves to identify the actual challenge and output commitment attached to the recovered source witnesses.

### C.10 Combined error terms

One admissible choice for the independent-row error bound, from Corollary 1, is

$$\epsilon_{P,i}^{\text{iid}} = \left( \frac{3\sigma_i^2}{3\sigma_i^2 - 1} \left( \frac{7}{8} + 2^{-1000} \right) \right)^{m_i}. \quad (56)$$

For a vector fixed before the complete seed, Lemma 7 gives

$$|\Pr[\text{PASS}_w(\Pi_{\text{seed}})] - \Pr[\text{PASS}_w(\Pi_{\text{iid}})]| \leq \eta, \quad \eta = 2^{-150}. \quad (57)$$

The selected profiles instead use the sharper 384-row bound described in Appendix C.5:

$$\epsilon_{P,i}^{\text{iid}} = 2^{-192}, \quad \epsilon_{P,i} = 2^{-192} + 2^{-150}. \quad (58)$$

Fresh seeds on successive attempts imply  $\hat{\epsilon}_{P,i} = 1 - (1 - \epsilon_{P,i})^{R_{P,i}} \leq R_{P,i} \epsilon_{P,i}$ .

For the augmented row count  $R_i$  of Section 5.5, put  $D_i^{\text{agg}} = \lceil R_i/4 \rceil - 1$ . Lemma 8 allows the aggregation bound

$$\epsilon_{\text{agg},i} = q^{-3} + (1 - q^{-3})D_i^{\text{agg}}/q^4. \quad (59)$$

The field check has conditional errors at most  $|\mathbb{K}|^{-1}$  for each equality-weight restriction and  $3|\mathbb{K}|^{-1}$  for each degree-three sumcheck round. Its combined bound is

$$\epsilon_F = 1 - (1 - |\mathbb{K}|^{-1})^{\ell_F} (1 - 3|\mathbb{K}|^{-1})^{\ell_F} \leq 4\ell_F/|\mathbb{K}|.$$

Substituting these terms and the five-point sampling loss into (26) and (27) gives the statistical bound used in Theorem 2.

## D Computational Reduction and Module-SIS Instantiation

### D.1 One assumption term per sampled matrix

The setup distribution is a product of uniform module-matrix distributions. Its coordinates are the front matrices, the three matrices  $A_i, B_i, D_i$  at every nonterminal layer, and the two matrices  $A_L, B_p$  at the pivot terminal. With three front matrices and  $L$  nonterminal layers, there are  $3L + 5$  coordinates. The  $k = 1024$  profile has  $L = 4$  and therefore  $J = 17$ . The  $k = 32$  profile has one additional front matrix because its auxiliary segment is split into two independently committed blocks.

A matrix can be used in several commitment equations. In particular,  $A_{\text{st}}$  authenticates  $k + 2$  state records: one accumulator,  $k$  fresh states, and one folded output. A short difference for any one of these uses solves the same  $A_{\text{st}}$  instance.

Role  $j$  has parameters  $\vartheta_j = (q, d, a_j, b_j, \beta_j)$ . Its Module-SIS game is defined in (4), with a uniform challenge  $A_j \in R_q^{a_j \times b_j}$  and advantage  $\text{Adv}_{\vartheta_j}^{\text{MSIS}}(t)$ . Every role uses the coefficient  $\ell_2$  norm and requires the returned kernel to be nonzero modulo  $q$ .

Coefficient expansion produces a structured negacyclic matrix of size  $da_j$  by  $db_j$ . It is an exact representation of the module equation and coefficient norm. Its entries are not an independently sampled scalar matrix. The attack estimates reported later use the specified coefficient-lattice model; the computational theorem retains the role-specific Module-SIS assumptions.

### D.2 Embedding a challenge without changing setup

Fix a role  $j$ . The reduction receives a Module-SIS matrix with exactly the dimensions and ring of that role. It places this matrix in coordinate  $j$  and independently samples every other coordinate from its declared distribution. Public

data derived from these matrices is then generated by the protocol’s usual algorithm. The reduction runs the adversary’s initialization to reach the pre-Cauchy checkpoint and invokes the truncated extractor.

Public statements and commitments may depend on the entire setup; they are generated after this embedding in the same order as in the real experiment. The full joint distribution, rather than only each marginal matrix distribution, is consequently preserved.

When the extractor returns a role- $j$  kernel, the reduction checks the equation, the norm, and modular nonzeroness, then returns the vector. Any other role or failure is a failure for this reduction. This defines a separate reduction for every fixed  $j$ .

To make the role events disjoint, the extractor returns immediately on the first verified kernel. Its role label is part of the output. If a local comparison can produce several kernels, use a fixed public order to select the first. Writing  $K_j$  for the event that the truncated extractor returns role  $j$ , we have

$$\Pr\left[\bigcup_j K_j\right] = \sum_j \Pr[K_j].$$

Each fixed-role reduction gives the corresponding bound on  $\Pr[K_j]$ .

### D.3 A single global clock and complete runtime charges

To keep the formulas readable, this appendix writes  $\bar{p}, P, B, E_B, S_j^{\max}$  for the seeded quantities  $\bar{p}_N, P_N, B_N, E_{B_N}, S_{j,N}^{\max}$  of Theorem 3. Fix the cost model before interpreting an advantage bound. The extractor’s clock covers its entire execution from the supplied pre-Cauchy checkpoint. It includes verifier sampling, prover continuations, copying and restoring complete states, nested extractor calls, field and ring arithmetic, integer norm checks, kernel validation, and output construction. The expectation  $P$  in Theorem 3 bounds this whole clock, averaged over the actual setup and initialized prover state.

Let  $T$  be that runtime and stop the full computation after  $B$  steps. Markov’s inequality gives  $\Pr[T > B] \leq P/B$ . The same clock runs across all layers and all recovery trials. There is no separate truncation loss per layer or per matrix.

For the fixed-role reduction, write

$$S_j^{\max} = S_{\text{keys},j}^{\max} + S_{\text{public},j}^{\max} + S_{\text{init},j}^{\max} + S_{\text{interface},j}^{\max}.$$

The terms bound, respectively: generation of the other setup matrices; generation of the remaining public data; the adversary’s initialization up to the supplied checkpoint; and any challenge conversion, input/output handling, or final return work outside the extractor’s clock. Any cost already inside that clock need not be counted a second time. A common larger bound is also valid.

With these charges, the solver runs in at most  $B + S_j^{\max}$ .

When the adversary is strict polynomial time and the protocol dimensions are polynomially bounded, the initialization and interface terms have worst-case

polynomial bounds. Exact finite-alphabet sampling can be charged as a primitive operation in a stated arithmetic model. In a random-bit model it requires the additional treatment below; an unbounded rejection loop is not a worst-case setup bound.

#### D.4 Quantifiers when averaging over setup

Let  $\omega$  contain the complete setup and initialized prover state, and let  $s(\omega)$  be its pre-Cauchy checkpoint. The statistical extractor is the same algorithm for all  $\omega$ . Its pointwise guarantee is

$$\Pr[\text{Src} \cup \text{Ker} \mid \omega] \geq \max\{0, p(s(\omega)) - \kappa_{\text{alg}}\}.$$

The same public error bound applies to every setup. Therefore

$$\Pr[\text{Src} \cup \text{Ker}] \geq \mathbb{E}_\omega p(s(\omega)) - \kappa_{\text{alg}} = \bar{p} - \kappa_{\text{alg}}.$$

The runtime assumption is separately  $\mathbb{E}_\omega \mathbb{E}[T \mid \omega] \leq P$ . It suffices for the single averaged Markov bound.

After truncation, subtracting the labelled kernel events gives

$$\begin{aligned} \Pr[E_B \text{ returns Src}] &\geq \bar{p} - \kappa_{\text{alg}} - P/B - \sum_j \Pr[K_j] \\ &\geq \bar{p} - \kappa_{\text{alg}} - P/B - \sum_j \text{Adv}_{\vartheta_j}^{\text{MSIS}}(B + S_j^{\max}). \end{aligned}$$

Taking the maximum with zero yields the stated theorem. In a parameter family with a growing number of roles, the assumption must control the entire matrix family uniformly, or directly bound its summed advantage. Negligibility for each fixed role index, with no uniform bound as the index grows, would not suffice.

#### D.5 Bounded setup sampling from random bits

A uniform  $\mathbb{F}_q$  coefficient can be sampled by drawing  $m_q = \lceil \log_2 q \rceil$  unbiased bits and accepting the integer if it is below  $q$ . Its acceptance probability is greater than  $1/2$ . For the extra keys in reduction  $j$ , let

$$N_{\text{set},j} = d \sum_{h \neq j} a_h b_h$$

be the number of required base-field coefficients. Bound every coefficient sampler by  $r_{\text{set}}$  attempts and fail the reduction on exhaustion. Then

$$\eta_{\text{set},j} \leq N_{\text{set},j} 2^{-r_{\text{set}}}. \quad (60)$$

If other public setup randomness uses rejection sampling, its bounded samplers and failure probabilities are included in the same accounting.

Couple this implementation with exact rejection sampling using the same random bits. On the event that no cap is reached, the two setups agree, including all derived public data. Successful sampled coefficients have the required uniform distribution. Thus

$$\Pr[K_j] \leq \text{Adv}_{\vartheta_j}^{\text{MSIS}}(B + S_j^{\max}) + \eta_{\text{set},j},$$

where  $S_j^{\max}$  now includes at most  $N_{\text{set},j} r_{\text{set}}$  coefficient-sampling attempts. Summing these bounds adds  $\sum_j \eta_{\text{set},j}$  to the theorem. Sampling inside the extractor itself remains covered by its full clock and the one global truncation.

## D.6 Mapping to coefficient-lattice estimates

The estimator maps a module matrix with  $a_j$  rows and  $b_j$  columns to coefficient dimensions  $n_{\text{SIS}} = da_j$  and  $m_{\text{SIS}} = db_j$ , preserving the Euclidean radius  $\beta_j$ . The classical and quantum cost conventions and their numerical outputs are specified in Appendix D.9.

## D.7 Concrete profiles

The arity profiles share the field, ring, semantic relation, short challenges, retry bounds, and canonical value encoding. The specialized compiler and message formats are used for the  $k = 1024$  profile. Their backend schedules are selected separately. Table 5 lists the common parameters and the selected matrix ranks.

The block partitions and radices for  $k \leq 32$  were selected from a finite grid with zero to five nonterminal layers followed by one pivot terminal. The block count ranges from 2 to 32 at each layer. The nonterminal radix is a power of two between 4 and 128. This initial search fixed the row ranks, projection parameters, challenge sets, retry caps, and pivot codec, retaining schedules that satisfied the modular projection range and the reference instances' column and kernel-radius bounds. The present profiles retain those block partitions and radices, with norm bounds and matrix ranks updated as described in Section 7. The accompanying artifact includes the search code and selected schedules.

For each nonterminal layer,  $A_i$  commits a response block,  $B_i$  commits the radix digits of its block commitments, and  $D_i$  commits symmetric-data digits. The pivot terminal uses  $A_L$  and  $B_p$ . Together with the front-end matrices, this gives five independent matrices for the terminal-only profiles, twelve for  $k = 32$ , and seventeen for  $k = 1024$ . Repeated state commitments use the same  $A_{\text{st}}$  and do not create additional assumption terms.

The response thresholds are given by (19) and (25). Kernel radii follow (28) and (29), with  $\lceil 2\gamma_0 \rceil$  for each front-end matrix.

## D.8 Statistical error

Table 8 reports the exact rational calculations through their displayed negative binary logarithms. The field term, Cauchy consistency term, projection term,

**Table 5.** Parameters shared by the selected profile families.

| Parameter                        | Common value                   |
|----------------------------------|--------------------------------|
| Base field modulus $q$           | $2^{48} - 59$                  |
| Ring $R_q$                       | $\mathbb{Z}_q[X]/(X^{64} + 1)$ |
| Extension degree and polynomial  | $4; \theta^4 - 4\theta^2 + 2$  |
| Semantic dimensions $(n, y, r)$  | $(69, 4, 4)$                   |
| Front-end row rank               | 4                              |
| Auxiliary and pivot row ranks    | 6 and 2                        |
| Squared norm slack $\sigma^2$    | 38 400/4 121                   |
| Projection rows                  | 384                            |
| Aggregation rows and challenge   | 3 rows; 42 bytes per layer     |
| Response codec                   | Fixed-padded signed Rice       |
| Short-challenge support          | $\{-2, -1, 0, 1, 2\}^{64}$     |
| Challenge operator bound; moment | 82; 128                        |
| Projection and response caps     | 160 and 160                    |
| Pivot radix and digit count      | 8 and 16                       |
| Main row rank                    | 16                             |

Both families use compact handoff vectors, poles  $0, \dots, k-1$ , and scales  $\lambda_i = 1$ . The  $k = 32$  profile splits its auxiliary commitment into independently sampled blocks of 4,350 and 3,696 ring columns. The 384-row projection specialization applies to every displayed profile.

**Table 6.** Selected finite schedules and norm bounds.

| $k$  | $i$ | $s_i$ | $\rho_i$ | $n_i$ | $N_i$     | $S_i$         | $G_i$           |
|------|-----|-------|----------|-------|-----------|---------------|-----------------|
| 2    | 0   | 3     | T        | 673   | 129 216   | 80 643        | 20 644 608      |
| 4    | 0   | 3     | T        | 981   | 188 352   | 95 765        | 24 515 840      |
| 8    | 0   | 4     | T        | 1 198 | 306 688   | 126 009       | 32 258 304      |
| 16   | 0   | 6     | T        | 1 415 | 543 360   | 186 497       | 47 743 232      |
| 32   | 0   | 9     | 64       | 1 765 | 1 016 640 | 307 473       | 78 713 087      |
|      | 1   | 6     | 64       | 961   | 369 024   | 260 916 513   | 66 794 627 327  |
|      | 2   | 5     | T        | 639   | 204 480   | 163 024 939   | 41 734 384 384  |
| 1024 | 0   | 30    | 64       | 2 711 | 5 205 120 | 4 250 457     | 1 088 116 991   |
|      | 1   | 14    | 128      | 1 459 | 1 307 264 | 1 153 671 816 | 295 339 984 895 |
|      | 2   | 8     | 128      | 837   | 428 544   | 1 282 739 299 | 328 381 260 543 |
|      | 3   | 6     | 128      | 555   | 213 120   | 626 217 742   | 160 311 741 951 |
|      | 4   | 5     | T        | 445   | 142 400   | 415 737 499   | 106 428 799 744 |

$N_i = 64s_i n_i$  includes alignment padding.  $S_i$  and  $G_i$  are squared-norm bounds. T denotes the pivot terminal, whose radix is 8. The profiles with  $k \leq 16$  enter the terminal directly;  $k = 32$  uses two nonterminal layers and  $k = 1024$  uses four.

structured-aggregation term, and coordinate-recovery term are evaluated separately before summation.

For the  $k = 1024$  profile, put

$$\epsilon_P = 2^{-192} + 2^{-150}, \quad \delta_F = 1 - (1 - q^{-4})^{23}(1 - 3q^{-4})^{23}.$$

**Table 7.** Complete projection-seed and selected-response lengths.

| $k$  | $i$ | Block bits | Depth | Seed bits | Seed bytes | $p_i$ bytes |
|------|-----|------------|-------|-----------|------------|-------------|
| 2    | 0   | 1 103      | 17    | 57 339    | 7 168      | 534         |
| 4    | 0   | 1 109      | 17    | 57 651    | 7 207      | 538         |
| 8    | 0   | 1 117      | 18    | 61 417    | 7 678      | 547         |
| 16   | 0   | 1 126      | 19    | 65 289    | 8 162      | 561         |
| 32   | 0   | 1 140      | 20    | 69 520    | 8 690      | 580         |
| 1024 | 1   | 1 183      | 18    | 65 047    | 8 131      | 815         |
|      | 2   | 1 199      | 17    | 62 331    | 7 792      | 795         |
|      | 0   | 1 180      | 22    | 79 038    | 9 880      | 673         |
|      | 1   | 1 212      | 20    | 73 912    | 9 239      | 866         |
|      | 2   | 1 256      | 18    | 69 062    | 8 633      | 867         |
|      | 3   | 1 249      | 17    | 64 931    | 8 117      | 838         |
|      | 4   | 1 175      | 17    | 61 083    | 7 636      | 825         |

The seed includes the initial block and every affine-Toeplitz hash description. Selected projection responses use the fixed-padded signed Rice codec. The independent structured aggregation challenge adds 42 bytes per layer.

**Table 8.** Statistical bounds and independent commitment-matrix counts.

| $k$  | $-\log_2 \kappa_{\text{alg}}$ | Independent matrices |
|------|-------------------------------|----------------------|
| 2    | 136.349232                    | 5                    |
| 4    | 136.349232                    | 5                    |
| 8    | 135.940489                    | 5                    |
| 16   | 135.361850                    | 5                    |
| 32   | 133.626152                    | 12                   |
| 1024 | 131.976186                    | 17                   |

The second column is the negative binary logarithm of an exact statistical upper bound. It is not a computational-security estimate. The accompanying artifact contains role-specific classical and quantum estimates for all displayed profiles.

Exact rational arithmetic gives

$$\begin{aligned} \kappa_{\text{alg}} \leq & \delta_F + \frac{2048}{q^4 - 1024} + 5[1 - (1 - \epsilon_P)^{160}] + \frac{5}{q^3} \\ & + (1 - q^{-3}) \frac{68\,618}{q^4} + 1600 \left( \frac{62}{5^{64}} + \frac{1}{5^{64} - 1} \right). \end{aligned} \quad (61)$$

The five layers have  $\sum_i D_i^{\text{agg}} = 68\,618$  and  $\sum_i s_i = 63$ , with one coordinate at the terminal sampled from  $\mathcal{D}_L \setminus \{0\}$ . These counts give the aggregation and coordinate-recovery terms in (61). The honest retry-exhaustion bound is  $10 \cdot 2^{-160}$ .

## D.9 Attack-estimation parameters

For a role with  $a$  rows and  $b$  columns over  $R_q$ , the estimator input is

$$(n_{\text{SIS}}, m_{\text{SIS}}, q, \beta, \text{norm}) = (64a, 64b, 2^{48} - 59, \beta, 2).$$

**Table 9.** Independent Module-SIS roles and attack estimates for the  $k = 1024$  profile.

| Role             | $a$ | $b$    | Radius $\beta$ | Classical      | Quantum        | BKZ block |
|------------------|-----|--------|----------------|----------------|----------------|-----------|
| $A_{\text{st}}$  | 4   | 219    | 12 587         | 231.318        | 216.806        | 717       |
| $A_H$            | 4   | 12 288 | 12 587         | 231.318        | 216.806        | 717       |
| $A_{\text{aux}}$ | 4   | 59 381 | 12 587         | 231.318        | 216.806        | 717       |
| $A_0$            | 16  | 2 711  | 2 827 705 411  | 167.895        | 159.888        | 485       |
| $B_0$            | 6   | 3 840  | 207 365        | 202.997        | 191.376        | 614       |
| $D_0$            | 6   | 11 160 | 207 365        | 202.997        | 191.376        | 614       |
| $A_1$            | 16  | 1 459  | 5 962 830 476  | <b>155.624</b> | <b>148.847</b> | 441       |
| $B_1$            | 6   | 1 568  | 218 658        | 201.042        | 189.617        | 607       |
| $D_1$            | 6   | 2 205  | 218 658        | 201.042        | 189.617        | 607       |
| $A_2$            | 16  | 837    | 4 166 255 944  | 161.201        | 153.865        | 461       |
| $B_2$            | 6   | 896    | 152 777        | 215.286        | 202.430        | 658       |
| $D_2$            | 6   | 756    | 152 777        | 215.286        | 202.430        | 658       |
| $A_3$            | 16  | 555    | 3 394 633 258  | 164.548        | 156.876        | 473       |
| $B_3$            | 6   | 672    | 124 482        | 224.228        | 210.474        | 690       |
| $D_3$            | 6   | 441    | 124 482        | 224.228        | 210.474        | 690       |
| $A_4$            | 16  | 445    | 139 003 366    | 236.648        | 221.743        | 731       |
| $B_p$            | 2   | 256    | 1 024          | 210.864        | 198.317        | 647       |

The module matrix has shape  $a \times b$ . The estimator input is coefficient Euclidean SIS with dimensions  $(64a, 64b)$ , modulus  $q = 2^{48} - 59$ , and radius  $\beta$ . Classical and quantum columns report  $\log_2(\text{rop})$  under the MATZOV models. Every listed block size is at most 1024. The calculation uses SageMath 10.9; the estimator version and individual outputs are supplied with the artifact.

The official `SIS.estimate` interface gives the heuristic coefficient-lattice costs in Table 9. Its classical and quantum MATZOV estimates lie within the documented block-size range. These modeled costs retain the role-specific Module-SIS assumptions and extraction-time loss in Theorem 3; they do not cover every attack exploiting module structure.

For  $k = 2$ , the estimator returns no finite attack estimate for  $A_H$  under either model. The remaining smaller-profile estimates exceed 128 in  $\log_2(\text{rop})$ , with 38 entries using block sizes above 1024. The artifact records these extrapolations and the nonfinite outputs.

## E Efficiency, Serialization, and Operation Counts

We first prove the finite-schedule bounds of Theorem 4, then give the serialization rules and operation counts for the concrete profiles.

### E.1 Proof of the finite-schedule bounds

*Proof.* Equation (36) gives

$$N_{i+1}^{\text{raw}} = \tilde{O}_\kappa(N_i^{1-\nu} + N_i^{2\nu} + 1) = \tilde{O}_\kappa(N_i^{1-\nu} + 1),$$

where the second equality uses  $2\nu \leq 1 - \nu$ . Iterating a constant number of layers yields  $N_i^{\text{raw}} = \tilde{O}_\kappa(N^{(1-\nu)^i} + 1)$ . The largest witness-dependent ring term is proportional to  $s_i N_i$ , giving the exponent  $1 + \nu$ . Sumcheck processes geometrically shrinking tables and adds  $O(2^{\ell_F})$  work beyond construction of the field relation.

A nonterminal layer sends two commitments, one selected projection response, and retry tags. The terminal sends the disclosed block data and one bounded response. Equation (36), together with  $2\nu \leq 1 - \nu$ , bounds the last-layer payload by  $\tilde{O}_\kappa(N^\gamma + 1)$ . In the other direction, a complete projection seed has polylogarithmic length, the structured aggregation challenge contains seven base-field elements, and the short challenge has  $s_i$  ring elements. The largest of these terms is  $\tilde{O}_\kappa(N^\nu + 1)$ . Public transposes, projection expansion, and field checks give the verifier-time bound.

## E.2 Canonical message syntax

A canonical element of  $\mathbb{F}_q$  occupies six little-endian bytes; encodings at least  $q$  are rejected. An element of  $\mathbb{K}$  lists its four base-field coefficients in the basis  $(1, \theta, \theta^2, \theta^3)$ , and a ring element lists the coefficients of  $1, X, \dots, X^{63}$ . These objects occupy 24 and 384 bytes, respectively.

A projection seed contains the initial block and every affine-Toeplitz hash description required by the fixed generator. Seed bits are packed least-significant bit first, with zero high padding. Consecutive generated bit pairs  $(u, v)$  encode the trit  $u - v$  in row-major order. The structured aggregation challenge sends  $\zeta \in \mathbb{K}$  and the three free coordinates of  $\chi \in \mathcal{H}$ , for 42 bytes. Short-challenge coefficients use three bits. Each message is byte-aligned separately, and the public profile fixes every vector length and retry cap.

For a centered integer response  $v \in \mathbb{Z}^n$  with public bound  $\sum_j v_j^2 \leq T$ , map

$$u(z) = \begin{cases} 2z, & z \geq 0, \\ -2z - 1, & z < 0. \end{cases}$$

With public Rice parameter  $a \geq 1$ , encode  $u$  by its unary quotient, a terminator, and an  $a$ -bit remainder. Put  $U = \lfloor T/2^{2a-2} \rfloor$ ,  $t = \lfloor \sqrt{U/n} \rfloor$ , and  $r = \min\{n - 1, \lfloor (U - nt^2)/(2t + 1) \rfloor\}$ . The large-arity profile uses the exact maximum bit length

$$L_{\max} = n(a + 1) + nt + r. \quad (62)$$

To see this, a quotient  $j$  requires coefficient magnitude at least  $2^{a-1}j$ , attained by the nonnegative coefficient with this value. For a fixed sum of quotients, their sum of squares is minimized when they differ by at most one. A sum  $nt + r$  therefore requires energy  $2^{2a-2}(nt^2 + r(2t + 1))$ . Inverting this expression gives the displayed capacity, attained by a vector with  $n - r$  quotients  $t$  and  $r$  quotients  $t + 1$ . The profile chooses the public  $a$  minimizing this length and pads to its byte ceiling. The reference profiles use the valid larger capacity  $n(a + 1) + \lfloor \sqrt{nT}/2^{a-1} \rfloor$ . Both parsers read exactly  $n$  coordinates, enforce the norm bound, and reject nonzero trailing padding.

**Table 10.** Communication accounting.

**(a) Directional communication and optional input transport**

| $k$  | $P \rightarrow V$ | $V \rightarrow P$ | All $k + 1$ incoming commitments | Maximum accepted interaction |
|------|-------------------|-------------------|----------------------------------|------------------------------|
| 2    | 66 565            | 8 170             | 4 608                            | 1 226 213                    |
| 4    | 83 775            | 8 257             | 7 680                            | 1 249 711                    |
| 8    | 107 546           | 8 752             | 13 824                           | 1 352 682                    |
| 16   | 146 990           | 9 332             | 26 112                           | 1 477 294                    |
| 32   | 128 569           | 26 251            | 50 688                           | 4 145 561                    |
| 1024 | 120 224           | 46 355            | 1 574 400                        | 7 325 872                    |

All entries are bytes. The directional columns give no-retry interaction. Incoming commitments are a separate optional transport cost. Maximum accepted interaction sets every applicable retry cap to 160 and is not an expectation. The CRS and application-level public-coordinate encoding are excluded.

**(b) Phase ledger for  $k = 1024$**

| Phase or message family   | $P \rightarrow V$ | $V \rightarrow P$ | Total          |
|---------------------------|-------------------|-------------------|----------------|
| New front-end commitments | 4 608             | 0                 | 4 608          |
| Cauchy challenge          | 0                 | 24                | 24             |
| Field subprotocol         | 1 176             | 1 104             | 2 280          |
| Nonterminal layer 0       | 5 283             | 10 642            | 15 925         |
| Nonterminal layer 1       | 5 476             | 9 617             | 15 093         |
| Nonterminal layer 2       | 5 477             | 8 867             | 14 344         |
| Nonterminal layer 3       | 5 448             | 8 303             | 13 751         |
| Pivot terminal            | 92 756            | 7 798             | 100 554        |
| <b>Total</b>              | <b>120 224</b>    | <b>46 355</b>     | <b>166 579</b> |

The table counts one accepted attempt at each layer using the serialization specified in this appendix.

### E.3 Field and carrier operations

The sparse-entry counts in Appendix B determine the matrix-application costs. A forward sparse application performs one  $\mathbb{K}$ -by- $\mathbb{F}_q$  multiplication per stored entry and accumulates the result in  $\mathbb{K}$ . For  $k = 1024$ , the three matrices contain 25 645 337 entries in total. A transpose multiplies the  $\mathbb{K}$ -valued row weight by each stored coefficient. Padding rows perform no arithmetic.

The equality table for  $2^{\ell_F}$  rows uses  $2^{\ell_F} - 1$  extension-field multiplications and the same number of subtractions. At  $k = 1024$ , each count is 8 388 607. The degree-three sumcheck then processes geometrically shrinking tables. The prover’s operation ledger keeps these  $\mathbb{K}$  additions and multiplications separate from sparse applications and from the twelve terminal base-field claims.

For the carrier algorithms of Table 2, the count expands each  $Q$ ,  $B$ , and  $J$  call in the arithmetic of the concrete relation, as specified in Appendix A. Multiplication by a base-field interpolation constant is a  $\mathbb{K}$ -by- $\mathbb{F}_q$  operation, while a product of two arbitrary extension-field values is a  $\mathbb{K}$  multiplication.

#### E.4 Projection, aggregation, and lattice work

One forward projection scan reads  $m_i N_i$  ternary entries. Summed across the selected layers, this is 208 650 240 entries for the selected  $k = 16$  profile and 2 801 836 032 entries for  $k = 1024$ . Retries and the three aggregation adjoints are charged separately. Those adjoints contribute  $3m_i N_i$  ternary-weighted terms and may share one generated stream. Seed generation counts affine-Toeplitz hash evaluations and binary convolution operations; these are not field or ring multiplications.

For an augmented relation with  $R_i$  rows, structured aggregation uses  $J_i - 1 = \lceil R_i/4 \rceil - 1$  general multiplications in  $\mathbb{K}$  to advance the powers of  $\zeta_i$ , together with linear basis operations for the four positions in each group. The  $k = 1024$  profile has  $\sum_i (J_i - 1) = 68\,618$ . Both parties perform the deterministic expansion. Applying the resulting rows to the public relation is counted through the structured forward and transpose operators, not through a dense  $3 \times R_i$  matrix.

The remaining backend ledger distinguishes full  $R_q$  products, products by bounded public coefficients, ring additions, radix decomposition, centered coefficient operations, and integer norm checks. Literal CRS storage is 16 203 264 bytes for the selected  $k = 16$  profile and 197 147 136 bytes for  $k = 1024$ .