Notes on BulletProofs++ Range Proofs

Philipp Jovanovic[1,2] and Arnab Roy[1]

[1]Mysten Labs
[2]University College London

Abstract

We review the Bulletproofs++ construction which replaces the bit-decomposition inner-product argument of Bulletproofs with a reciprocal set-membership argument over base-16 digits. We present a self-contained specification of the 64-bit range proof as well as a general batched variant parametric in the batch size M and bit width b and prove honest-verifier zero-knowledge and exact-form knowledge soundness. We describe the evaluation and optimizations of our Rust implementation over Ristretto255, which proves up to 7.8× and verifies up to 4.5× faster than regular Bulletproofs. Finally, we correct two formulas of the original paper: the affine T^3 term of the scalar polynomial $p_s(T)$ carries a factor of 2, and the slot-0 component of the blinding vector r_S carries the opposite sign.

1 Introduction

A range proof certifies that a committed value lies in a public interval without revealing the value. In confidential transaction systems the range proofs attached to outputs dominate both transaction size and verification cost, so constant-factor improvements in proof length and verifier work translate directly into system-level savings. Bulletproofs [1] established the current baseline: proof size logarithmic in the bit width, no trusted setup, and a verifier that reduces to a single multi-scalar multiplication. A 64-bit Bulletproof occupies 672 bytes (16 group elements and 5 scalars) and its verifier multi-scalar multiplication runs over 147 bases.

Bulletproofs++ [2] improves on both figures by changing the encoding of the range statement. Rather than decompose the value into bits and prove each bit binary through an inner-product argument, it decomposes the value into base-16 digits and proves digit membership through a reciprocal argument: under a verifier challenge α the prover commits to the inverses $(\alpha + d_k)^{-1}$ of the digits d_k, and a single partial-fraction identity in α certifies that every digit d_k lies in $\{0, \ldots, 15\}$. The resulting constraint system is then proven by a weighted norm-linear argument, which commits to one witness vector where the inner-product argument commits to two. For a 64-bit range the proof is 416 bytes (10 group elements and 3 scalars) and the verifier multi-scalar multiplication runs over 36 bases.

These savings come at the cost of a construction whose correctness is concentrated in a single object: a Laurent polynomial $f(T)$ in the evaluation challenge, which must vanish identically in every honest execution. The constraint vectors, the public polynomials and the blinding of the post-challenge commitment are each defined by the coefficients of f they must produce or cancel, one condition per power of T (Section 3). If any one of these formulas is wrong, the polynomial produced by an honest prover fails to vanish in the corresponding coefficient, and the verifier rejects honest proofs. The original paper states these formulas at the generality of an arbitrary arithmetic circuit, which leaves a gap between the published description and a working implementation.

Contributions. We give a specification of the 64-bit reciprocal range proof in which every formula is stated concretely for the parameters at hand and has been validated against a from-scratch Rust implementation over Ristretto255 producing 416-byte proofs. We then present a generic batched construction parametric in the batch size M and the bit width b, of which the single-value 64-bit protocol is the instance $(M, b) = (1, 64)$, and prove it perfect honest-verifier zero-knowledge and knowledge sound in the exact form (Section 5): the inputs are taken as bare group elements, and the extractor returns an opening $V_i = v_i G + s_i H_0$ with v_i in range for each of them. We describe the optimizations our implementation[1] applies to the prover and the verifier and report measurements against Bulletproofs across bit widths and batch sizes (Section 6); on the configurations measured, proving is between 1.7 and 7.8 times faster and verification between 1.1 and 4.5 times faster depending on the instances to be proven. Finally, we document two corrections to the formulas of [2] that our implementation exposed (Section 7): the affine T^3 term of the scalar polynomial $p_s(T)$ carries a factor of 2, matching the doubling of every witness term in the tested identity, and the slot-0 component of the blinding vector r_S carries the opposite sign to the published substitution. Under either published formula the polynomial $f(T)$ fails to vanish on honest witnesses, and the verifier rejects honest proofs.

Organization. Section 2 fixes the notation, the hardness assumption and the commitment key. Section 3 gives an overview of the construction, parametric in the batch size and bit width, and motivates its building blocks. Section 4 states the relation for a single 64-bit value and gives the protocol that proves it: the message flow (Section 4.2), the digit decomposition and the reciprocal argument (Section 4.3), the commitment layout (Section 4.4), the constraint vectors, the public polynomials and the evaluation at the challenge point (Sections 4.5–4.7), the cancellation of the error terms by the blinding vector r_S (Section 4.8), and the norm-linear argument together with the proof-size account (Section 4.9). Section 5 gives the general batched construction and proves it zero-knowledge and knowledge sound. Section 6 reports the implementation, its optimizations and its evaluation against Bulletproofs, and Section 7 states the two errata.

2 Preliminaries

This section fixes the notation and parameters used throughout and states the common reference string / commitment key; the key and the range relation of Section 3 correspond to [BP++ §6.4, Eq. (72)]. References to the original paper are written [BP++ §X] or [BP++ Eq. (Y)] and point to the paper version [2] listed in the bibliography.

Notation. We denote by $\mathbb{F}_q$ the field with q elements and $\mathbb{F}_q^\times$ its multiplicative group. Vectors are indexed from zero, written $x = (x_0, \ldots, x_{n-1}) \in \mathbb{F}_q^n$, and $\|$ denotes concatenation. For indices $a \le b$ we write $x_{a:b} = (x_a, \ldots, x_b)$ for a subvector and $x_{a:} = x_{a:n-1}$ for a suffix. For $x, y \in \mathbb{F}_q^n$ we write $\langle x, y\rangle = \sum_k x_k y_k$ for the inner product and $x \circ y$ for the entry-wise product; the same bracket denotes the multi-scalar multiplication $\langle x, \mathbf{P}\rangle = \sum_k x_k P_k$ when the second argument is a vector of group elements. Polynomial arguments are carried in a formal variable T, and the combined polynomials of Sections 4.6 and 4.8 are Laurent polynomials, that is, elements of $\mathbb{F}_q[T, T^{-1}]$ with finitely many nonzero terms. For such an f we write $f[T^p]$ for the coefficient of T^p.

Parameters. The construction is parametric in a batch size $M \ge 1$ and a bit width b with $4 \mid b$. Each value has $D = b/4$ base-16 digits, the batch has $N_d = MD$ digits in total, and the norm

[1] `https://github.com/MystenLabs/fastcrypto/tree/main/fastcrypto/src/bulletproofspp`

length is $n_m = \min\{2^\kappa : 2^\kappa \geq \max(N_d, 16)\}$, the smallest power of two that accommodates both the digits and the sixteen digit values. The single-value 64-bit instance has $(M, b) = (1, 64)$ and $D = N_d = n_m = 16$.

Group and hardness assumption. We denote by $\mathbb{G}$ a cyclic group of prime order q, written additively, in which the discrete logarithm problem is hard. We require $2^b \ll q$ and $M(n_m{+}8) \ll q$, so that no integer up to $\max(N_d, 16)$ vanishes modulo q and q exceeds every branch count of the extractor (Section 5). Soundness is stated relative to the discrete logarithm relation assumption: for uniformly sampled $P_1, \ldots, P_n \in \mathbb{G}$ it is infeasible to produce a nonzero $a \in \mathbb{F}_q^n$ with $\sum_k a_k P_k = 0$. A nontrivial relation of this form yields a discrete logarithm, so the assumption follows from hardness of the discrete logarithm problem in $\mathbb{G}$.

Commitment key. The public parameters are $(G, \mathbf{H}, \mathbf{G}) \in \mathbb{G} \times \mathbb{G}^8 \times \mathbb{G}^{n_m}$ with the base G carrying scalar values, $\mathbf{H} = (H_0, \ldots, H_7)$ carrying the blinding at $H_0, \ldots, H_6$ and the linear witness at H_7 (so $N_v = 1$ in the notation of the original paper), and $\mathbf{G} = (G_0, \ldots, G_{n_m-1})$ carrying the norm witness at one base per base-16 digit of the batch. This split mirrors the norm-linear commitment structure of [BP++ §4, Eq. (15)]. All generators are sampled transparently by hashing to the group, so the scheme requires no trusted setup and finding a relation among the generators is as hard as computing discrete logarithms.

Pedersen commitment. A value $v \in \mathbb{F}_q$ is committed as $V = vG + sH_0$ with the blinding factor s sampled uniformly from $\mathbb{F}_q$. The commitment is perfectly hiding, since sH_0 is uniform in $\mathbb{G}$, and binding under the relation assumption above.

Weighted inner product and weighted norm. For a weight $\mu \in \mathbb{F}_q^\times$ and $x, y \in \mathbb{F}_q^n$,

$$\langle x, y \rangle_\mu = \sum_{k=0}^{n-1} x_k y_k \mu^{k+1}, \qquad \|x\|_\mu^2 = \langle x, x \rangle_\mu. \tag{1}$$

The weighted inner product is bilinear and symmetric, and writing $\bar{\mu} = (\mu, \mu^2, \ldots, \mu^n)$ it satisfies $\langle x, y \rangle_\mu = \langle x \circ \bar{\mu}, y \rangle$, which is the form the implementation evaluates (Section 6). The weight is derived from a verifier challenge ρ as $\mu = \rho^2$ and is squared in each round of the folding argument. Replacing the inner product of Bulletproofs by this norm is what allows the prover to commit to one witness vector instead of two.

Norm-linear commitment. Every prover message is a commitment of the form $C = \sigma G + \langle \ell, \mathbf{H} \rangle + \langle n, \mathbf{G} \rangle$ with $\sigma \in \mathbb{F}_q$, $\ell \in \mathbb{F}_q^8$ and $n \in \mathbb{F}_q^{n_m}$, where n is the argument of the weighted norm and ℓ the argument of the linear part. Section 4.4 fixes which witness occupies which slot, and Section 4.9 states the argument that proves the norm-linear relation.

Arguments of knowledge. The protocol of Section 3, stated in full in Sections 4 and 5, is a public-coin interactive argument: every verifier message is a field element sampled uniformly and independently of the prover's messages. Such an argument is *complete* if the honest prover convinces the honest verifier with probability 1; *knowledge sound* if there is an expected polynomial-time extractor that, given rewinding access to a prover succeeding with non-negligible probability, outputs a witness for the relation being proved or a discrete logarithm relation on the key; and *perfect honest-verifier zero knowledge* if there is a simulator that, on the statement and uniformly

sampled challenges, produces transcripts distributed identically to real ones. The extractor of Section 5 follows the standard route of building a tree of accepting transcripts by forking the verifier challenges. The argument is made non-interactive by the Fiat–Shamir heuristic, deriving each challenge by hashing the transcript so far, which preserves knowledge soundness and yields zero-knowledge in the random oracle model. The implementation uses Merlin transcripts[2] for this purpose.

3 Overview

3.1 Statement

The prover P demonstrates, for $M \geq 1$ Pedersen commitments $V_1, \ldots, V_M$, knowledge of openings $(v_1, s_1) \ldots, (v_M, s_M)$ to a verifier V such that each value v_i has at most b bits with (random) blinders s_i. The corresponding range relation $\mathcal{R}_{M,b}$ is defined as:

$$\mathcal{R}_{M,b} = \left\{ \left(V_1, \ldots, V_M, (v_i, s_i)_{i=1}^{M} \right) : V_i = v_i G + s_i H_0, 0 \leq v_i < 2^b \right\}. \tag{2}$$

Section 4 states every formula of the construction for the instance $(M, b) = (1, 64)$, and Section 5 gives the general construction and proves it secure.

To do so, we (1) express the range statement as an arithmetic circuit and (2) reduce satisfiability of that circuit to a single instance of the norm-linear relation (see below), which the norm-linear argument (NLA) then proves. We first provide an overview of the NLA since that will inform the construction of the arithmetic circuit.

3.2 Norm-linear argument

The NLA proves that a commitment $C = \sigma G + \langle \ell, \mathbf{H} \rangle + \langle n, \mathbf{G} \rangle$ opens to vectors ℓ, n and a scalar σ satisfying the norm-linear relation $\sigma = \langle c, \ell \rangle + \|n\|_\mu^2$ for a public vector c. The NLA proof is logarithmic in the combined length of ℓ and n. However, it provides compression only and no zero-knowledge. The latter thus has to be ensured through the arithmetic circuit encoding. Everything specific to range proofs happens in the arithmetic circuit encoding before the NLA is applied: both parties construct an instance (C, c) of the norm-linear relation from the inputs V_i, the prover's commitments and the verifier's challenges, such that a prover who knows openings of all V_i to values in range knows an opening of C satisfying the norm-linear relation, while any other prover knows such an opening only with negligible probability over the challenges. The weighted norm of n is the only quadratic term, so all circuit witnesses are placed in n, where products of witnesses and their pairings with public constraint vectors arise as cross terms, while ℓ enters linearly and holds the blinding.

3.3 Arithmetic circuit encoding

The arithmetic circuit encoding proceeds in three steps, after which an evaluation challenge hands the resulting instance to the NLA.

First, the range statement is rewritten as an arithmetic circuit. The prover decomposes every v_i into D base-16 digits, so that the batch has N_d digits d_k, one per entry of $\mathbf{G}$, and the range statement becomes one value-link row $v_i = \sum_t 16^t d_{i,t}$ per input, one constraint of the circuit per row, together with the membership of every digit in $\{0, \ldots, 15\}$. The reciprocal argument [BP++ §3.2] checks all

[2] `https://github.com/dalek-cryptography/merlin`

N_d memberships at once: for a verifier challenge α, the identity $\sum_k (\alpha + d_k)^{-1} = \sum_j m_j (\alpha + j)^{-1}$, with m_j the number of digits equal to j, holds whenever every digit lies in the set and fails with overwhelming probability otherwise. The prover therefore commits to the digits and the multiplicities as C_L and C_O before α is drawn, then computes the reciprocals $r_k = (\alpha + d_k)^{-1}$ and commits to them as C_R. Clearing denominators in the definition of r_k gives the Hadamard rows $d_k r_k = 1 - \alpha r_k$, the only quadratic rows of the circuit and the reason a norm is needed, and the identity itself is the set-membership row. Furthermore, since the V_i are given as bare group elements, shape rows assert that all their coordinates outside the (G, H_0)-plane vanish, so that the extractor returns a Pedersen opening of each (Section 4.3).

Second, all rows are combined into a single coefficient of the norm linear polynomial $f(T) = \sigma(T) - \langle c(T), \ell(T) \rangle - \|n(T)\|_\mu^2$ in variable T, where $(\sigma(T), \ell(T), n(T))$ is the opening of a combined commitment $C(T)$ into which the prover's commitments, the inputs V_i combined by powers of a verifier challenge λ, and the public polynomials enter each at its own power of T, and public vector polynomial $c(T)$. As we will see, $f(T)$ is a Laurent polynomial over powers T^{-2} to T^{10}. The encoding is constructed in such a way so that the cross terms of the weighted norm land at predictable powers, and the public constraint vectors are placed so that every pairing of a witness with its constraint lands at T^3 (Sections 4.5 and 4.6). The T^3 coefficient is a weighted sum of all rows, with one weight per row derived from the verifier challenges ρ and λ. It therefore vanishes whenever every row is satisfied and in particular for an honest witness. Conversely, if some row is violated, the coefficient is a nonzero polynomial in ρ and λ, which vanishes only with negligible probability over the verifier's random choice. Two further challenges β and δ separate the blinding and C_O from the witness.

Third, privacy comes from the encoding, since the NLA reveals its final vectors in the clear. Every commitment carries random blinding on G and $\mathbf{H}$, and the last commitment C_S carries fresh masks n_S, ℓ_S that make the revealed vectors uniformly distributed. Blinding and masks produce cross terms in $f(T)$, which must still vanish identically. For an honest prover, whose inputs have no coordinates outside the (G, H_0)-plane and whose blinding follows a fixed zero pattern, $f(T)$ contains terms in T^{-2} to T^6 only. The zero pattern also keeps the cross terms out of the T^3 coefficient, which must depend on the witness alone since it is the coefficient that certifies the rows, and the eight free components of the blinding r_S of C_S are chosen to cancel the eight remaining coefficients, one each (Section 4.8), so that $f(T) \equiv 0$ for an honest prover.

Finally, the verifier sends an evaluation challenge τ, and substituting $T = \tau$ turns $f(\tau) = 0$ into the norm-linear relation on $C(\tau)$, a public linear combination of the prover's commitments, the inputs and the public polynomials (Section 4.7). Since τ is drawn after all commitments are fixed and f has at most thirteen coefficients, $f(\tau) = 0$ implies $f \equiv 0$ except with negligible probability over τ, so acceptance of the NLA on $(C(\tau), c(\tau))$, with the prover holding the evaluated opening $(\sigma(\tau), \ell(\tau), n(\tau))$, certifies every row.

4 Range proofs for a 64-bit value

This section discusses in detail the construction of Section 3 for a single 64-bit value, i.e., for the case $(M, b) = (1, 64)$.

4.1 Statement

The 64-bit range relation is defined as

$$\mathcal{R}_{64} = \big\{ \big(V; (v, s)\big) : V = vG + sH_0, 0 \le v < 2^{64} \big\}, \tag{3}$$

So the value v is deconstructed into 16 base-16 digits, one per entry of $\mathbf{G}$, and a single value is committed, so $N_v = 1$. Section 4.2 gives the message flow of the protocol. The remaining subsections then define the messages and the verifier's computations in protocol order, except that the blinding commitment of step 5 is treated after the evaluation of step 6, on which it depends.

4.2 Protocol flow and transcript

The interactive protocol follows [BP++ §5.3] (inner protocol) with the reciprocal-argument extension of [BP++ §6.2]. In practice it is made non-interactive via the Fiat–Shamir heuristic (using Merlin transcripts in the implementation).

Step	Direction	Data	Cost
1	$\mathsf{P} \to \mathsf{V}$	C_L, C_O	$2\mathbb{G}$
2	$\mathsf{P} \leftarrow \mathsf{V}$	α	
3	$\mathsf{P} \to \mathsf{V}$	C_R	$1\mathbb{G}$
4	$\mathsf{P} \leftarrow \mathsf{V}$	$\rho, \lambda, \beta, \delta$	
5	$\mathsf{P} \to \mathsf{V}$	C_S	$1\mathbb{G}$
6	$\mathsf{P} \leftarrow \mathsf{V}$	τ	
7	P and V run the NLA		$6\mathbb{G} + 3\mathbb{F}$

The prover starts by committing to the digits of v and their multiplicities as C_L and C_O (step 1). Only then does the verifier draw the reciprocal challenge α (step 2), so the prover cannot choose digits to match a known α. The prover computes the reciprocals $r_k = (\alpha + d_k)^{-1}$ and commits to them as C_R (step 3); Sections 4.3 and 4.4 give the constraints these three commitments must satisfy and their layout. The four challenges of step 4 serve different roles: ρ sets the weight $\mu = \rho^2$ of the norm, λ weights the linear constraints, β weights the blinding, and δ separates C_O from the other commitments in the combined polynomial; from them the verifier computes the public constraint vectors and polynomials of Sections 4.5 and 4.6. With these fixed, the prover sends the blinding commitment C_S (step 5), whose masks make the vectors revealed by the NLA uniformly distributed and whose blinding is chosen to cancel the error terms of $f(T)$ (Section 4.8). The verifier draws the evaluation point τ (step 6) and combines the commitments and the public polynomials into the commitment $C(\tau)$ and the public vector $c(\tau)$ of Section 4.7, while the prover computes the corresponding opening. Finally, they run the NLA (step 7, Section 4.9), which convinces the verifier that $C(\tau)$ opens to vectors satisfying the norm-linear relation, that is, $f(\tau) = 0$. Since τ is random, this implies that $f(T)$ vanishes identically except with negligible probability, in particular its T^3 coefficient, which encodes the constraints of Section 4.3.

The transcript consists of the four commitments C_L, C_O, C_R, C_S, the folding messages (X, R) of the three NLA rounds, and the final vectors $\ell \in \mathbb{F}_q^1$ and $n \in \mathbb{F}_q^2$ sent in the clear (Section 4.9). All challenges are derived via the (strong) Fiat–Shamir transform and are not transmitted: the random oracle is domain-separated by the full statement, that is, the public parameters, the bit width, the batch size and the commitments V_i, so that every statement is proven under its own independent random oracle. Overall the transcript is of size $10\mathbb{G} + 3\mathbb{F} = 416$ bytes at 32 bytes per element on a 256-bit curve, matching [BP++ Table 1, row 1×64].

4.3 Constraints

In this section we discuss how the range relation is encoded into constraints.

Digit decomposition. Write $v = \sum_{k=0}^{15} d_k \cdot 16^k$ with each $d_k \in \{0, \ldots, 15\}$. The set-membership check $d_k \in \{0, \ldots, 15\}$ is enforced by the reciprocal argument.

Reciprocal witnesses. After receiving a random challenge α, the prover computes $r_k = (\alpha + d_k)^{-1}$ for $k = 0, \ldots, 15$. Clearing the denominator converts the rational-function check into a Hadamard (entry-wise product) constraint.

Multiplicities. Let $m_j = |\{k : d_k = j\}|$ for $j = 1, \ldots, 15$ denote the *multiplicities*. The multiplicity of 0 is implicit: $m_0 = 16 - \sum_{j=1}^{15} m_j$. In the *inline multiplicities* layout [BP++ §6.4.2, Eq. (75)], the m_j are committed as part of C_O rather than in a separate commitment, which saves one group element.

Input shape. Since V is a bare group element, we name its coordinates over the whole key,

$$V = uG + \sum_{m=0}^{7} h_m H_m + \langle n_V, \mathbf{G} \rangle,$$

with an honest input being one with $u = v$, $h_0 = s$, and every other coordinate zero. Alongside the range circuit rows, the constraint system tests one *shape row* per coordinate of V outside the (G, H_0)-plane (Section 4.5), forcing all of n_V and $h_1, \ldots, h_7$ to vanish and enabling the extractor to return an opening of the form $vG + sH_0$.

Constraint families. In BP++ we work with four families of constraints. The first three are the rows of the arithmetic circuit that the reciprocal range proof encodes [BP++ §6.4.2, Theorem 4] and the fourth is the shape family introduced above, which has no counterpart in the original paper.

(a) **Hadamard** (16 rows): These constraints ensure that each committed r_k is the reciprocal $(\alpha + d_k)^{-1}$ of the committed digit d_k, so that the set-membership row is evaluated on the true digits rather than on values of the prover's choosing. This translates to:

$$d_k \cdot r_k = 1 - \alpha r_k \text{ for } k = 0, \ldots, 15. \tag{4}$$

(b) **Value link** (1 row): This constraint ensures that the committed digits recompose to the committed value, so that bounding every digit bounds v. This translates to:

$$\sum_{k=0}^{15} 16^k d_k = v. \tag{5}$$

(c) **Set membership** (1 row): This constraint ensures that every digit lies in $\{0, \ldots, 15\}$, since for random α the sum of the reciprocals matches the multiplicity-weighted sum over the set only in that case. This is the reciprocal vanishing check from [BP++ Eq. (11)]. This translates to:

$$\sum_{k=0}^{15} r_k + \sum_{j=1}^{15} \frac{j}{\alpha(\alpha+j)} m_j = \frac{16}{\alpha}. \tag{6}$$

(d) **Shape** (23 rows): These constraints ensure that the extracted opening of V is a Pedersen commitment $vG + sH_0$, i.e., that every coordinate of V outside the (G, H_0)-plane vanishes. This translates to:

$$n_{V,k} = 0 \text{ for } k = 0, \ldots, 15 \wedge h_m = 0 \text{ for } m = 1, \ldots, 7. \tag{7}$$

If every d_k belongs to $\{0, \ldots, 15\}$, the partial-fraction decomposition of $\sum_k (\alpha + d_k)^{-1}$ telescopes to the right-hand side of (6). A digit outside $\{0, \ldots, 15\}$ fails this check for random α except with negligible probability.

4.4 Commitment structure

The commitment layout follows [BP++ §5.2.1, Eq. (36)]. Every prover message C_X, where $X \in \{L, O, R, S\}$ labels the four commitments of the protocol, is a *norm-linear commitment*

$$C_X = r_{X,0} G + \langle r_{X,1:7} \| \ell_X, \mathbf{H} \rangle + \langle n_X, \mathbf{G} \rangle, \tag{8}$$

with $r_X \in \mathbb{F}_q^8$ (blinding), $\ell_X \in \mathbb{F}_q^1$ (witness on $\mathbf{H}$), $n_X \in \mathbb{F}_q^{16}$ (witness on $\mathbf{G}$). G carries a scalar value, $\mathbf{G}$ carries the norm witness (digits, reciprocals, multiplicities), and $\mathbf{H}$ carries the linear witness plus blinding.

Witness placement (inline multiplicities). The inline-multiplicities variant [BP++ §6.4.2, Eq. (75)] places the multiplicities, the witness of the set-membership row, in the norm slot n_O of C_O, which the generic circuit reserves for witness elements appearing only in linear constraints, avoiding a separate commitment:

	$n_X \in \mathbb{F}_q^{16}$	$\ell_X \in \mathbb{F}_q^1$	$r_X \in \mathbb{F}_q^8$
C_L	$(d_0, \ldots, d_{15})$	(0)	below
C_O	$(m_1, \ldots, m_{15}, 0)$	(0)	below
C_R	$(r_0, \ldots, r_{15})$	(0)	below
C_S	$n_S \xleftarrow{\$} \mathbb{F}_q^{16}$	$\ell_S \xleftarrow{\$} \mathbb{F}_q^1$	below

Blinding zero pattern. The blinding vectors r_O, r_L, r_R are committed in steps 1 and 3 of the interactive protocol between the prover and the verifier (Section 4.2), before β is drawn. They have zeros at fixed positions [BP++ §5.2.4] so that the blinding contributes nothing to the T^3 coefficient of $f(T)$. The prover later cancels the remaining error terms with r_S, which cannot reach the value power T^3 since $\hat{c}_r$ has no T^4 monomial (Section 4.7).

$$\begin{aligned}
r_O &= (r'_{O,0}, r'_{O,1}, r'_{O,2}, r'_{O,3}, 0, r'_{O,4}, r'_{O,5}, 0), \\
r_L &= (r'_{L,0}, r'_{L,1}, r'_{L,2}, 0, r'_{L,3}, r'_{L,4}, 0, 0), \\
r_R &= (r'_{R,0}, r'_{R,1}, 0, r'_{R,2}, r'_{R,3}, 0, 0, 0).
\end{aligned}$$

The zeros at positions $r_{O,4}$, $r_{L,3}$, $r_{R,2}$ are the ones that prevent blinding from leaking into the T^3 coefficient. The remaining zero positions ($r_{L,6}$, $r_{L,7}$, $r_{R,5}$, etc.) cap the T-support of the blinding part of $f(T)$ at T^6, matching the eight error rows that r_S cancels (Section 4.8).

Rescaled input. $\hat{V} = 2V$, with coordinates $\hat{v} = 2u$, $r_V = (0, 2h_0, 2h_1, \ldots, 2h_6)$, $\ell_{\hat{V}} = 2h_7$, $n_{\hat{V}} = 2n_V$; for an honest input $\hat{v} = 2v$, $r_V = (0, 2s, 0, \ldots, 0)$, and $\ell_{\hat{V}}$, $n_{\hat{V}}$ vanish. The factor of 2 arises from the norm expansion $\|p_n + \hat{n}\|_\mu^2 = \|p_n\|_\mu^2 + 2\langle p_n, \hat{n} \rangle_\mu + \|\hat{n}\|_\mu^2$; the cross-term carries a factor of 2, so the input commitment is doubled to match.

4.5 Constraint vectors

This subsection corresponds to [BP++ §5.2.2, Eqs. (40)–(44)], extended by the shape rows on the coordinates of V, which have no counterpart in the original paper.

Polynomial identity. For the norm-linear argument (Section 4.9) to accept, the full polynomial

$$f(T) = \sigma(T) - \big\langle c(T), \ell(T) \big\rangle - \big\| n(T) \big\|_\mu^2$$

must vanish identically as a polynomial in T, not only at $T = \tau$. Since $n(T)$ and $\ell(T)$ run from T^{-1} to T^3 and $c(T)$ from T^{-1} to T^7 (Section 4.7), $f(T)$ is a Laurent polynomial with support T^{-2} to T^{10} for arbitrary openings. For the honest prover, the blinding zero pattern of Section 4.4 and the vanishing off-plane coordinates (7) of the input cut the support to T^{-2} to T^6. The constraint vectors below are designed so that the T^3 coefficient, the *value term*, is zero for any valid witness, which is the soundness condition. The other eight coefficients, the *error terms*, are generically non-zero, since the random blinding in r_O, r_L, r_R contributes to them. The prover cancels them with r_S (Section 4.8).

Row weights. The constraint vectors are public: the verifier computes them from the challenges alone. They encode the four constraint families of Section 4.3 into the value term. Each linear row receives its own power of λ, so that this one coefficient separates all of them: the value link sits at λ^0, set membership at λ, and the shape rows at λ^2 through λ^{24}. Throughout, $\mu = \rho^2$ and $\bar{\mu} = (\mu, \mu^2, \ldots, \mu^{16})$.

Witness vectors. The vectors $c_{n,L}$, $c_{n,R}$, $c_{n,O}$ pair with the committed witness vectors n_L, n_R, n_O. They follow from substituting (4)–(6) into the randomized constraint combination [BP++ Eq. (40)] and reading off the coefficient of each witness vector. The division by μ^{k+1} compensates the weight that the norm (1) places on slot k. The value link (5) gives:

$$[c_{n,L}]_k = \frac{16^k}{\mu^{k+1}}, \text{ for } k = 0, \ldots, 15. \tag{9}$$

The reciprocals appear in set membership (6), weighted by λ, and in the Hadamard rows (4), which the norm already weights by μ^{k+1} and which contribute the constant α:

$$[c_{n,R}]_k = \frac{\lambda}{\mu^{k+1}} + \alpha, \text{ for } k = 0, \ldots, 15. \tag{10}$$

The multiplicities appear only in set membership, with coefficient $j/(\alpha(\alpha + j))$ at $j = k + 1$; the last slot is zero because m_0 is implicit:

$$[c_{n,O}]_k = \frac{(k{+}1)\lambda}{\alpha(\alpha{+}k{+}1)\mu^{k+1}}, \text{ for } k = 0, \ldots, 14; \qquad [c_{n,O}]_{15} = 0. \tag{11}$$

Affine terms. The affine terms are the contributions of the constraints that multiply no witness vector, coming from the right-hand sides of (5) and (6):

$$\lambda_{a_l} = \frac{16\lambda}{\alpha}, \qquad \mu_{a_m} = \sum_{k=1}^{16} \mu^k. \tag{12}$$

Here λ_{a_l} is the right-hand side $16/\alpha$ of (6) scaled by λ, and μ_{a_m} is the sum of the Hadamard right-hand sides, each equal to 1, weighted by $\bar{\mu}$.

Shape rows. Each coordinate of V outside the (G, H_0)-plane receives its own λ-power, continuing the ladder:

$$[c_{n,V}]_k = \frac{\lambda^{k+2}}{\mu^{k+1}}, \text{ for } k = 0, \ldots, 15; \qquad c_{\ell,V} = \big(0,\ \lambda^{18},\ \lambda^{19},\ \lambda^{20},\ \lambda^{21},\ \lambda^{22},\ \lambda^{23} \parallel \lambda^{24}\big). \tag{13}$$

$c_{n,V}$ pairs the **G**-coordinates $n_{\hat{V}}$ of the input and $c_{\ell,V}$ its $H_1, \ldots, H_7$ coordinates, so the tested coefficient contains each of the 23 off-plane coordinates at its own λ-power, and the extractor forces all of them to vanish (Section 5). The zero in the first entry of $c_{\ell,V}$ leaves the H_0-coordinate unpaired: the blinding s is the one coordinate the shape rows must not touch. For the honest prover every shape row is zero, so the shape rows cost nothing, in proof size or in prover work beyond the public computation.

Remark on signs. The signs of the constraint vectors (9)–(13) differ from [BP++ Eqs. (42)–(43)] by a global sign flip on the linear-constraint row. Both conventions produce a zero value term, since the challenge λ absorbs the sign. The convention used here keeps every term positive in the implementation.

4.6 Public polynomials

The public polynomials $p_n(T)$ and $p_s(T)$ are the verifier's contribution to the combined polynomial $f(T)$. They correspond to [BP++ §5.2.3, Eqs. (46)–(48)], extended by the shape blocks. Both prover and verifier compute them from the public challenges alone.

Norm polynomial. The norm polynomial $p_n(T)$ collects the constraint vectors into a single vector-valued polynomial in T, placed by the rule that a vector entering $C(\tau)$ at T^p meets its constraint block at T^{3-p}, so that every row lands in the value coefficient:

$$p_n(T) = c_{n,V} + Tc_{n,R} + T^2 c_{n,L} + \delta^{-1} T^3 c_{n,O} \in \mathbb{F}_q^{16}[T]. \tag{14}$$

The input enters at T^3, so its shape block $c_{n,V}$ sits at T^0.

Scalar polynomial. The scalar polynomial $p_s(T)$ is the G-coordinate of the public part of $C(\tau)$. It carries the norm of $p_n(T)$, which cancels against the public part of $\|n(T)\|_\mu^2$, and the affine right-hand sides (12) of the constraints at the value power T^3:

$$p_s(T) = \big\|p_n(T)\big\|_\mu^2 + 2\big(\lambda_{a_l} + \mu_{a_m}\big)T^3. \tag{15}$$

The factor of 2 on the affine T^3 term is required (Section 7). It matches the factor of 2 that the norm expansion $\|p_n + \hat{n}\|_\mu^2 = \|p_n\|_\mu^2 + 2\langle p_n, \hat{n}\rangle_\mu + \|\hat{n}\|_\mu^2$ places on the cross-terms at T^3. Without it, $f[T^3] \neq 0$ for valid witnesses.

The value coefficient vanishes identically for the honest witness: $\|p_n\|_\mu^2$ cancels between $p_s(T)$ and $\|n(T)\|_\mu^2$, the witness pairings $2\langle c_{n,L}, n_L\rangle_\mu + 2\langle c_{n,R}, n_R\rangle_\mu + 2\langle c_{n,O}, n_O\rangle_\mu + 2\langle n_L, n_R\rangle_\mu$ reproduce $2(\lambda_{a_l} + \mu_{a_m}) + \hat{v}$ by the Hadamard, value-link and set-membership families, every shape pairing is zero because the off-plane coordinates of an honest input are (7), and the blinding contributes nothing at T^3 by the zero pattern of Section 4.4. For arbitrary openings the full T^3 coefficient is displayed in (29).

4.7 Polynomial evaluation at τ

After the verifier sends τ, both sides evaluate the combined polynomial at $T = \tau$ to obtain concrete vectors and a scalar that are fed into the norm-linear argument. This corresponds to [BP++ §5.2.3, Eqs. (49)–(57)].

Combined commitment. The verifier forms the combined commitment $C(\tau)$ as a linear combination of the prover's commitments at powers of τ, plus the public polynomials evaluated at τ:

$$C(\tau) = P(\tau) + \tau^{-1}C_S + \delta C_O + \tau C_L + \tau^2 C_R + \tau^3 \hat{V}, \tag{16}$$

where $P(\tau) = p_s(\tau)G + \langle p_n(\tau), \mathbf{G}\rangle$. By construction, $C(\tau)$ is a norm-linear commitment $\sigma(\tau)G + \langle \ell(\tau), \mathbf{H}\rangle + \langle n(\tau), \mathbf{G}\rangle$ whose three coordinates are the evaluated witnesses below.

Norm vector. The norm vector $n(\tau)$, the $\mathbf{G}$-coordinate of $C(\tau)$, is the sum of the public polynomial $p_n(\tau)$ and the committed witness vectors, each at the power of τ at which its commitment enters (16):

$$n(\tau)_k = \underbrace{[c_{n,V}]_k + \tau[c_{n,R}]_k + \tau^2[c_{n,L}]_k + \tau^3\delta^{-1}[c_{n,O}]_k}_{\text{public part } p_n(\tau)_k} + \underbrace{\tau^{-1}[n_S]_k + \delta[n_O]_k + \tau[n_L]_k + \tau^2[n_R]_k + \tau^3[n_{\hat{V}}]_k}_{\text{witness part}}, \tag{17}$$

with $n_{\hat{V}} = 0$ for an honest input.

Linear vector. The linear vector $\ell(\tau)$, the $\mathbf{H}$-coordinate of $C(\tau)$, combines the blinding vectors in slots 1 to 7 with the linear witness in slot 8, at the same powers of τ:

$$\ell(\tau) = \big(r_{1:7}(\tau)\big\|\tau^{-1}\ell_S + \tau^3\ell_{\hat{V}}\big), \text{ where } r_i(\tau) = \tau^{-1}r_{S,i} + \delta r_{O,i} + \tau r_{L,i} + \tau^2 r_{R,i} + \tau^3 r_{V,i}.$$

For an honest input $\ell_{\hat{V}} = 0$.

Constraint vector. The public constraint vector $c(\tau)$ adds the shape constants $c_{\ell,V}$ (13) to the blinding monomials $\hat{c}_r(\tau)$ of [BP++ §5.2.3, Eq. (55)]:

$$c(\tau) = \hat{c}_r(\tau) + c_{\ell,V} = \big(\beta\tau^{-1}, \beta\tau + \lambda^{18}, \beta\tau^2 + \lambda^{19}, \beta\tau^3 + \lambda^{20}, \beta\tau^5 + \lambda^{21}, \beta\tau^6 + \lambda^{22}, \beta\tau^7 + \lambda^{23}\big\|\lambda^{24}\big).$$

The β-monomials pair the blinding into the error rows at the correct T-powers, and the λ-constants pair the τ^3 input coordinates $r_{V,2:7}$ and $\ell_{\hat{V}}$ into the value row. Two gaps are essential. $\hat{c}_r$ has no T^4 monomial: the blinding zero pattern of Section 4.4 already ensures no error at T^3, and a T^4 entry would pair the mask $\tau^{-1}r_S$ into the value row. And the first entry of $c_{\ell,V}$ is zero: slot 1 carries the input's H_0-coordinate $r_{V,1} = 2s$, the one coordinate the shape rows must leave free.

Scalar. The scalar $\sigma(\tau)$, the G-coordinate of $C(\tau)$, collects the scalar polynomial, the rescaled value and the combined blinding scalar:

$$\sigma(\tau) = p_s(\tau) + \hat{v}\tau^3 + r_0(\tau).$$

4.8 Cancelling the error terms

This step corresponds to [BP++ §5.2.4, Eqs. (58)–(60)] and [BP++ §5.3]. The value term of $f(T)$ vanishes for an honest witness by the design of the constraint vectors, but the eight error terms do not, since the random blinding of C_O, C_L, C_R and the masks of C_S contribute to them. The prover therefore chooses the blinding r_S of C_S so as to cancel these eight coefficients. Such a choice always exists because the eight components of r_S enter $f(T)$ triangularly: each of the slots 1 to 7 is the only unknown at one error power, and slot 0 is the only unknown remaining at T^{-1} once the others have been fixed.

Control of the error terms. Since C_S enters the combined commitment (16) with coefficient τ^{-1}, the component $r_{S,i}$ appears in $\ell(\tau)$ as $\tau^{-1} r_{S,i}$ for $i = 1, \ldots, 7$. Pairing it with the β-monomial $\beta\tau^{a_i}$ of $c(\tau)$, where a_i denotes the τ-power of the i-th entry of $\hat{c}_r(\tau)$, yields the term

$$(\beta\tau^{a_i}) \cdot (\tau^{-1} r_{S,i}) = \beta r_{S,i} \tau^{a_i - 1},$$

so that along this route each component reaches a distinct T-power of $f(T)$. The λ-constants of $c_{\ell,V}$ open a second route, along which the masks $\tau^{-1} r_{S,2:7}$ and $\tau^{-1}\ell_S$ are paired into the T^{-1} coefficient and into no other. The component $r_{S,0}$ behaves differently, as it enters $f(T)$ through $r_0(T)$ in $\sigma(T)$ rather than through $\langle c, \ell \rangle$. Its contribution at T^{-1} is $+T^{-1} r_{S,0}$, so that its divisor in the solution below is -1 rather than β.

The error polynomial. To isolate the quantities the prover must cancel, we split $f(T)$ into the *error polynomial* $\hat{f}(T)$, the part of $f(T)$ that involves neither $r_0(T)$ nor $\langle c(T), \ell(T) \rangle$, and the remainder:

$$f(T) = \hat{f}(T) + r_0(T) - \big\langle c(T), \ell(T) \big\rangle, \qquad \hat{f}(T) = p_s(T) + \hat{v}T^3 - \big\| n(T) \big\|_\mu^2.$$

Here $n(T)$ denotes the full norm vector of Section 4.7, including the mask term $T^{-1} n_S$ that is sampled together with C_S before r_S is determined. Writing $n(T) = T^{-1} n_S + \hat{n}(T)$, the mask contributes $T^{-2}\|n_S\|_\mu^2 + 2T^{-1}\langle n_S, \hat{n}(T) \rangle_\mu$, which occupies the powers T^{-2} to T^2 and leaves T^3 untouched. Consequently $\hat{f}(T)$ has support T^{-2} to T^6, and its T^3 coefficient coincides with the value term of $f(T)$, which vanishes for valid witnesses; the two remaining terms of the split contribute nothing at T^3, by the blinding zero pattern of Section 4.4 for the committed blinding and by (7) for the input coordinates paired by the λ-constants. The eight remaining coefficients of the error polynomial, at the powers $\{-2, -1, 0, 1, 2, 4, 5, 6\}$, are the error terms.

The blinding equation. Rearranging the split, the condition $f(T) = 0$ becomes

$$\hat{f}(T) = \big\langle c(T), \ell(T) \big\rangle - r_0(T),$$

which coincides with [BP++ Eq. (58)] except that $c(T) = \hat{c}_r(T) + c_{\ell,V}$ replaces the blinding monomials $\hat{c}_r(T)$ alone, the shape constants $c_{\ell,V}$ contributing the λ-route. Here $\hat{c}_{r,j}(T) = \beta T^{a_j}$ are the monomials of $\hat{c}_r(T)$ and $r_i(T)$ is as in Section 4.7. The right-hand side is a polynomial in T whose coefficients depend linearly on r_S and on the previously committed r_O, r_L, r_R, r_V, so that the equation can be solved coefficient by coefficient.

Solving for r_S. Separating the terms that depend on r_S from those determined by r_O, r_L, r_R, r_V yields a triangular system. For $j \geq 1$, slot j pairs its monomial $\hat{c}_{r,j} = \beta T^{a_j}$ with the entry $T^{-1} r_{S,j}$ of $r_j(T)$ and thereby controls the power $T^{a_j - 1}$, whereas slot 0 enters through $-r_0(T)$ at T^{-1} and is solved last, absorbing the contributions of the λ-route. The correspondence is summarised in the following table.

r_S **slot**	**cancels**	**via**	**divisor**
$r_{S,0}$	T^{-1}	$-r_0(T)$	-1
$r_{S,1}$	T^{-2}	$\hat{c}_{r,1} = \beta T^{-1}$	β
$r_{S,2}$	T^{0}	$\hat{c}_{r,2} = \beta T^{1}$	β
$r_{S,3}$	T^{1}	$\hat{c}_{r,3} = \beta T^{2}$	β
$r_{S,4}$	T^{2}	$\hat{c}_{r,4} = \beta T^{3}$	β
$r_{S,5}$	T^{4}	$\hat{c}_{r,5} = \beta T^{5}$	β
$r_{S,6}$	T^{5}	$\hat{c}_{r,6} = \beta T^{6}$	β
$r_{S,7}$	T^{6}	$\hat{c}_{r,7} = \beta T^{7}$	β

To state the solution, let $\hat{r}_j(T) = \delta r_{O,j} + T r_{L,j} + T^2 r_{R,j} + T^3 r_{V,j}$ denote the committed part of slot j and $\hat{\ell}(T) = \big(\hat{r}_{1:7}(T) \| T^3 \ell_{\hat{V}}\big)$ the committed part of $\ell(T)$, so that $\ell(T) = T^{-1}(r_{S,1:7} \| \ell_S) + \hat{\ell}(T)$. The *residual polynomial*

$$\bar{f}(T) = \hat{f}(T) - \big\langle c(T), \hat{\ell}(T) \big\rangle + \hat{r}_0(T)$$

is the part of $f(T)$ that involves neither r_S nor ℓ_S and is therefore fixed by the time the prover forms C_S. The blinding equation then yields

$$\begin{aligned} r_{S,j} &= \frac{\bar{f}[T^{p_j}]}{\beta}, \qquad (p_1, \ldots, p_7) = (-2, 0, 1, 2, 4, 5, 6), \\ r_{S,0} &= -\Big(\bar{f}[T^{-1}] - \sum_{m=1}^{6} \lambda^{17+m} r_{S,m+1} - \lambda^{24} \ell_S \Big), \end{aligned} \tag{18}$$

which is (31) at $(M, b) = (1, 64)$. The equations for slots $1, \ldots, 7$ are diagonal, and the subtracted sums in the equation for slot 0 are the λ-route contributions of the masks already solved for.

Sign of $r_{S,0}$. The divisor -1 of slot 0 deserves comment. The substitution [BP++ Eq. (59)] correctly leaves the slot-0 component undivided by β but assigns it the opposite sign (Section 7).

4.9 Norm-linear argument

This subsection corresponds to [BP++ §4, Eqs. (15)–(26)]. In step 7 of the protocol (Section 4.2) the prover convinces the verifier that $C(\tau)$ opens to $(\sigma(\tau), \ell(\tau), n(\tau))$ satisfying the weighted norm-linear relation

$$\sigma(\tau) = \big\langle c(\tau), \ell(\tau) \big\rangle + \big\| n(\tau) \big\|_{\mu}^2, \qquad C(\tau) = \sigma(\tau) G + \langle \ell(\tau), \mathbf{H} \rangle + \langle n(\tau), \mathbf{G} \rangle, \tag{19}$$

where $\| \cdot \|_{\mu}^2$ is the weighted norm (1). We drop the argument τ in the remainder of this subsection.

Folding. Each round halves the lengths of ℓ and n by the symmetric fold of [BP++ §4.1, Eqs. (16)–(17)]. After padding ℓ, n, c, $\mathbf{H}$, $\mathbf{G}$ to even length, write $[\cdot]_0$ and $[\cdot]_1$ for the even-indexed and odd-indexed halves of a vector. The prover sends the two group elements

$$X = v_x G + \langle [\ell]_1, [\mathbf{H}]_0 \rangle + \langle [\ell]_0, [\mathbf{H}]_1 \rangle + \langle \rho [n]_1, [\mathbf{G}]_0 \rangle + \langle \rho^{-1} [n]_0, [\mathbf{G}]_1 \rangle,$$
$$R = v_r G + \langle [\ell]_1, [\mathbf{H}]_1 \rangle + \langle [n]_1, [\mathbf{G}]_1 \rangle,$$

with $v_x = 2\rho^{-1}\langle [n]_0, [n]_1 \rangle_{\mu^2} + \langle [c]_0, [\ell]_1 \rangle + \langle [c]_1, [\ell]_0 \rangle$ and $v_r = \|[n]_1\|^2_{\mu^2} + \langle [c]_1, [\ell]_1 \rangle$, which collect the cross terms and the odd-half terms of the relation [BP++ Eqs. (19)–(22)]; the letters X and R follow BP++ and are unrelated to the commitment labels of Section 4.4. The verifier replies with a challenge γ, and both parties reduce the instance by [BP++ Eqs. (18) and (23)]:

$$\begin{aligned} C &\leftarrow C + \gamma X + (\gamma^2 - 1) R, \\ \ell' &= [\ell]_0 + \gamma [\ell]_1, \\ n' &= \rho^{-1} [n]_0 + \gamma [n]_1, \\ c' &= [c]_0 + \gamma [c]_1, \\ \mathbf{G}' &= \rho [\mathbf{G}]_0 + \gamma [\mathbf{G}]_1, \\ \mathbf{H}' &= [\mathbf{H}]_0 + \gamma [\mathbf{H}]_1, \\ \rho &\leftarrow \mu, \quad \mu \leftarrow \mu^2, \end{aligned}$$

and the prover sets $\sigma' = \langle c', \ell' \rangle + \|n'\|^2_{\mu'}$ for the new weight $\mu' = \mu^2$. The reduced instance satisfies (19) whenever the original one does, and the linear independence of $1, \gamma, \gamma^2 - 1$ gives a round extractor for the converse.

Base case. The reduction is repeated as long as $\mathrm{len}(\ell) + \mathrm{len}(n) \geq 6$, since a round costs $2\mathbb{G}$ and halves the number of scalars that would otherwise be sent. The prover then sends ℓ and n in the clear, and the verifier computes $\sigma = \langle c, \ell \rangle + \|n\|^2_{\mu}$ from them and checks $C = \sigma G + \langle \ell, \mathbf{H} \rangle + \langle n, \mathbf{G} \rangle$ on the folded generators [BP++ §4.3, Eq. (25)]. Note that [BP++ §4.2] states the stopping condition as $\mathrm{len}(\ell) + \mathrm{len}(n) \leq 6$, which for the 64-bit instance would stop one round earlier and cost $8\mathbb{G} + 6\mathbb{F}$; the sizes in [BP++ Table 1] correspond to the rule used here.

Round trace. For the 64-bit range proof the NLA starts from $\ell \in \mathbb{F}_q^8$ and $n \in \mathbb{F}_q^{16}$ and runs three rounds, as the following table shows.

Round	$\mathrm{len}(\ell)$	$\mathrm{len}(n)$	**sum**	**Action**
(start)	8	16	24	fold
1	4	8	12	fold
2	2	4	6	fold
3	1	2	3	< 6: stop

The three rounds contribute $6\mathbb{G}$ and the base case $3\mathbb{F}$, which together with the four commitments C_L, C_O, C_R, C_S gives the transcript size of $10\mathbb{G} + 3\mathbb{F}$ stated in Section 4.2.

5 General batched and exact form construction

This section presents the general batched exact-form reciprocal range proof, parametric in the batch size $M \geq 1$ and the bit width b with $4 \mid b$, and proves its two security properties: perfect honest-verifier zero-knowledge outside a challenge set of density $O(1)/q$ (Theorem 1), and knowledge soundness in the exact form (Theorem 2), where the extractor outputs, for every input commitment, an opening $V_i = v_i G + s_i H_0$ with v_i in range, under no assumption on how the V_i were formed. Exactness rests on the shape rows of Section 3, aggregated by powers of λ into the same tested coefficient as the circuit rows, at no cost in proof size. A single transcript covers the whole batch, and the single-value 64-bit proof is the instance $(M, b) = (1, 64)$; both proofs are uniform in (M, b).

5.1 Setting

Parameters, group, key. Fix a batch size $M \geq 1$ and a bit width b with $4 \mid b$. The derived quantities and the commitment key are

$$\begin{aligned} &D = \tfrac{b}{4}, && N_d = MD, && n_m = \min\{2^\kappa \ : \ 2^\kappa \geq \max(N_d, 16)\}, \\ &G \in \mathbb{G}, && \mathbf{H} = (H_0, \ldots, H_7) \in \mathbb{G}^8, && \mathbf{G} = (G_0, \ldots, G_{n_m - 1}) \in \mathbb{G}^{n_m}, \end{aligned} \tag{20}$$

the digits per value, the total digit count, the norm length, and the transparently sampled generators. No independence assumption is made on their discrete logarithms. The field must dominate the parameters: no integer in $\{1, \ldots, \max(N_d, 16)\}$ vanishes modulo q, and q exceeds every branch count of Table 1 (both follow from $2^b \ll q$ and $M(n_m{+}8) \ll q$). For $\mu \in \mathbb{F}_q^\times$ write $\langle x, y \rangle_\mu = \sum_k x_k y_k \mu^{k+1}$ and $\|x\|_\mu^2 = \langle x, x \rangle_\mu$.

Statement. The prover demonstrates, for M commitments $V_1, \ldots, V_M$ received as bare group elements, knowledge of openings $V_i = v_i G + s_i H_0$ with $0 \leq v_i < 2^b$. Since the V_i are arbitrary group elements, we name their coordinates over the whole key, and record the aggregate through which they enter the protocol:

$$\begin{aligned} &V_i = u_i G + \sum_{m=0}^{7} h_{i,m} H_m + \big\langle n_{V_i}, \mathbf{G} \big\rangle, \qquad i = 1, \ldots, M, \\ &\hat{V} = 2 \sum_{i=1}^{M} \lambda^{i-1} V_i, \qquad \hat{v} = 2 \sum_{i=1}^{M} \lambda^{i-1} u_i, \qquad r_{V,1} = 2 \sum_{i=1}^{M} \lambda^{i-1} h_{i,0}, \end{aligned} \tag{21}$$

the last two being the G- and H_0-coordinates of $\hat{V}$. An input is honest iff $u_i = v_i$, $h_{i,0} = s_i$, and every other coordinate vanishes.

Transcript. Seven steps: the prover sends C_L, C_O; the verifier draws α; the prover sends C_R; the verifier draws $\rho, \lambda, \beta, \delta$ (set $\mu = \rho^2$); the prover sends C_S; the verifier draws τ; both run the norm-linear argument. Every prover message is a norm-linear commitment, and the blinding of the three witness commitments carries a fixed *zero pattern*:

$$\begin{aligned} C_X &= r_{X,0} G + \big\langle r_{X,1:7} \| \ell_X, \ \mathbf{H} \big\rangle + \big\langle n_X, \ \mathbf{G} \big\rangle, \\ r_O &= (r_{O,0}, r_{O,1}, r_{O,2}, r_{O,3}, 0, r_{O,5}, r_{O,6}, 0), \\ r_L &= (r_{L,0}, r_{L,1}, r_{L,2}, 0, r_{L,4}, r_{L,5}, 0, 0), \\ r_R &= (r_{R,0}, r_{R,1}, 0, r_{R,3}, r_{R,4}, 0, 0, 0), \end{aligned} \tag{22}$$

with $r_X \in \mathbb{F}_q^8$, $\ell_X \in \mathbb{F}_q$, $n_X \in \mathbb{F}_q^{n_m}$. The zeros at $r_{O,4}, r_{L,3}, r_{R,2}$ keep blinding out of the value row; the remaining zeros bound the T-support of the blinding. The honest witness placement is: n_L holds the digit blocks, $n_{L,D(i-1)+t} = d_{i,t}$ where $v_i = \sum_t d_{i,t} 16^t$; n_R the reciprocals $r_{i,t} = (\alpha + d_{i,t})^{-1}$ in the same slots; n_O the shared multiplicities $m_j = |\{(i,t) : d_{i,t} = j\}|$ in slots $0, \ldots, 14$ (m_0 implicit); slots $\geq N_d$ of n_L, n_R are zero; $\ell_L = \ell_O = \ell_R = 0$; and C_S carries fresh masks $n_S \xleftarrow{\$} \mathbb{F}_q^{n_m}$, $\ell_S \xleftarrow{\$} \mathbb{F}_q$ with a solved blinding r_S (31).

Public data. From the challenges alone the verifier forms the constraint blocks, one per committed vector, placed by the rule that a vector entering $C(\tau)$ at T^p meets its block at T^{3-p}, together with the shape exponents and the affine terms:

$$
\begin{aligned}
[c_{n,L}]_{D(i-1)+t} &= \frac{\lambda^{i-1} 16^t}{\mu^{D(i-1)+t+1}}, \qquad i = 1, \ldots, M, \quad t = 0, \ldots, D{-}1; \qquad [c_{n,L}]_k = 0 \;\; (k \geq N_d), \\
[c_{n,R}]_k &= \frac{\lambda^M}{\mu^{k+1}} + \alpha \;\; (k < N_d); \qquad [c_{n,R}]_k = 0 \;\; (k \geq N_d), \\
[c_{n,O}]_k &= \frac{(k{+}1)\lambda^M}{\alpha(\alpha{+}k{+}1)\mu^{k+1}} \;\; (k \leq 14); \qquad [c_{n,O}]_k = 0 \;\; (k \geq 15), \\
[c_{n,V}]_k &= \frac{\lambda^{M(k+1)+1}}{\mu^{k+1}}, \qquad k = 0, \ldots, n_m{-}1, \\
c_{\ell,V} &= \big(0,\ \lambda^{g_1},\ \lambda^{g_2},\ \lambda^{g_3},\ \lambda^{g_4},\ \lambda^{g_5},\ \lambda^{g_6} \;\big\|\; \lambda^{g_7}\big), \qquad g_m = M(n_m{+}m) + 1, \\
\lambda_{a_l} &= \frac{N_d \lambda^M}{\alpha}, \qquad \mu_{a_m} = \sum_{k=1}^{N_d} \mu^k.
\end{aligned}
\tag{23}
$$

It assembles them into the public polynomials

$$
\begin{aligned}
p_n(T) &= c_{n,V} + T c_{n,R} + T^2 c_{n,L} + \delta^{-1} T^3 c_{n,O}, \\
p_s(T) &= \big\|p_n(T)\big\|_\mu^2 + 2\big(\lambda_{a_l} + \mu_{a_m}\big) T^3, \\
c(T) &= \hat{c}_r(T) + c_{\ell,V}, \\
\hat{c}_r(T) &= \big(\beta T^{-1},\ \beta T,\ \beta T^2,\ \beta T^3,\ \beta T^5,\ \beta T^6,\ \beta T^7 \;\big\|\; 0\big).
\end{aligned}
\tag{24}
$$

Two gaps are essential: $\hat{c}_r$ omits T^4, and the first entry of $c_{\ell,V}$ is zero. Neither the post-challenge message C_S nor the honest blinding slot is therefore paired into the value row.

Combined commitment. After τ is drawn, the verifier forms, and the prover opens,

$$
\begin{aligned}
C(\tau) &= P(\tau) + \tau^{-1} C_S + \delta C_O + \tau C_L + \tau^2 C_R + \tau^3 \hat{V}, \\
P(\tau) &= p_s(\tau) G + \big\langle p_n(\tau),\ \mathbf{G} \big\rangle.
\end{aligned}
\tag{25}
$$

Coordinate by coordinate, $C(\tau)$ opens to the evaluation at $T = \tau$ of the Laurent vectors

$$
\begin{aligned}
n(T) &= T^{-1} n_S + \delta n_O + T n_L + T^2 n_R + T^3 n_{\hat{V}} + p_n(T), \\
\ell(T) &= \big(\ell_1(T), \ldots, \ell_7(T) \;\big\|\; T^{-1}\ell_S + T^3 \ell_{\hat{V}}\big), \\
\ell_j(T) &= T^{-1} r_{S,j} + \delta r_{O,j} + T r_{L,j} + T^2 r_{R,j} + T^3 r_{V,j}, \\
\sigma(T) &= p_s(T) + \hat{v} T^3 + r_0(T), \qquad r_0(T) = T^{-1} r_{S,0} + \delta r_{O,0} + T r_{L,0} + T^2 r_{R,0},
\end{aligned}
\tag{26}
$$

whose input-side coordinates are the aggregates $n_{\hat{V}} = 2\sum_i \lambda^{i-1} n_{V_i}$, $r_{V,j} = 2\sum_i \lambda^{i-1} h_{i,j-1}$ ($j = 1, \ldots, 7$; slot j of the key is H_{j-1}), and $\ell_{\hat{V}} = 2\sum_i \lambda^{i-1} h_{i,7}$.

The tested identity. The norm-linear argument establishes that $C(\tau)$ opens to $(\sigma(\tau), \ell(\tau), n(\tau))$ with

$$\sigma(\tau) = \big\langle c(\tau), \ell(\tau) \big\rangle + \big\| n(\tau) \big\|_\mu^2, \tag{27}$$

by $\nu = \max(3, \log_2 n_m - 2)$ rounds of symmetric folding, each side zero-padded to even length before halving ($C \leftarrow C + \gamma X + (\gamma^2 - 1)R$; $\ell' = [\ell]_0 + \gamma[\ell]_1$; $n' = \rho^{-1}[n]_0 + \gamma[n]_1$; $c' = [c]_0 + \gamma[c]_1$; $\mathbf{G}' = \rho[\mathbf{G}]_0 + \gamma[\mathbf{G}]_1$; $\mathbf{H}' = [\mathbf{H}]_0 + \gamma[\mathbf{H}]_1$; $\rho \leftarrow \mu$, $\mu \leftarrow \mu^2$) followed by a clear opening. For the honest prover, (27) is the evaluation at τ of a Laurent polynomial identity:

$$f(T) = \sigma(T) - \big\langle c(T), \ell(T) \big\rangle - \big\| n(T) \big\|_\mu^2 \equiv 0. \tag{28}$$

The value row. The T^3 coefficient of f collects every circuit row. For *arbitrary* openings of C_O, C_L, C_R and arbitrary input coordinates (21),

$$\begin{aligned} f[T^3] = &\sum_{i=1}^{M} \lambda^{i-1} L_i + E_0 + \lambda^M S \\ &- 4 \sum_{k=0}^{n_m-1} \sum_{i=1}^{M} \lambda^{M(k+1)+1+(i-1)} n_{V_i,k} - 2 \sum_{m=1}^{7} \sum_{i=1}^{M} \lambda^{g_m+(i-1)} h_{i,m}, \end{aligned} \tag{29}$$

where the three scalar rows are

$$\begin{aligned} L_i &= 2u_i - 2 \sum_{t=0}^{D-1} 16^t n_{L,D(i-1)+t} - 4\delta \big\langle n_O, n_{V_i} \big\rangle_\mu, \\ E_0 &= -\beta\big(\delta r_{O,4} + r_{L,3} + r_{R,2}\big) - 2 \sum_{k=0}^{N_d-1} \mu^{k+1} \big(n_{L,k} n_{R,k} + \alpha n_{R,k} - 1\big) - 2 \sum_{k=N_d}^{n_m-1} \mu^{k+1} n_{L,k} n_{R,k}, \\ S &= 2\Big(\frac{N_d}{\alpha} - \sum_{k=0}^{N_d-1} n_{R,k} - \sum_{j=1}^{15} \frac{j}{\alpha(\alpha+j)} n_{O,j-1}\Big). \end{aligned} \tag{30}$$

The identity is verified by expanding $\|n(T)\|_\mu^2$ and $\langle c(T), \ell(T) \rangle$ at T^3: the μ^{k+1}-normalization of $c_{n,L}, c_{n,V}$ cancels the weighted pairing, the block indexing of $c_{n,L}$ regroups the digit slots by value, and the aggregate coordinates $n_{\hat{V}}, r_{V,1:7}, \ell_{\hat{V}}$ expand into per-input coordinates, one power of λ each. Every row family occupies its own monomials: the value links at $\lambda^0, \ldots, \lambda^{M-1}$ (the degree-0 row contaminated by E_0), set membership at λ^M, and the $M(n_m{+}7)$ shape rows at $\lambda^{M+1}, \ldots, \lambda^{M(n_m+8)}$, one input coordinate per power, without collision. For the honest witness every row vanishes: $L_i = 0$ by the value link and $n_{V_i} = 0$; $E_0 = 0$ by the Hadamard rows $d_{i,t} r_{i,t} = 1 - \alpha r_{i,t}$ on digit slots, the zero padding, and the blinding zero pattern; $S = 0$ by the partial-fraction telescoping $\sum_{i,t} (\alpha + d_{i,t})^{-1} = N_d/\alpha - \sum_j m_j j/(\alpha(\alpha+j))$ of the shared multiplicities; and the shape sums vanish since honest inputs have no coordinates outside the (G, H_0)-plane.

Design invariant. No component of the post-challenge message C_S appears in (29). Its components enter the Laurent vectors (26) at T^{-1}; the β-monomials of $\hat{c}_r$ pair them into T^{a-1} with $a \in \{-1, 1, 2, 3, 5, 6, 7\}$, the T^4 gap excluding T^3; the constant entries of $c_{\ell,V}$ pair them into T^{-1};

and within the norm they meet the placements $T^{-1}, \dots, T^3$ at $T^{-2}, \dots, T^2$. The coefficients of (29) are therefore fixed by the messages preceding the step-4 challenges. The extractor of Section 5.3 forks λ against those branch-independent coefficients, and the honest prover cancels the error rows after seeing all of step 4.

Choosing r_S. The remaining coefficients of f, at powers $p \in \{-2, -1, 0, 1, 2, 4, 5, 6\}$ (the *error rows*), are generically nonzero; the prover cancels them by the choice of r_S. Write $\hat{f}(T) = p_s(T) + \hat{v}T^3 - \|n(T)\|_\mu^2$, let $\hat{r}_j(T) = \delta r_{O,j} + T r_{L,j} + T^2 r_{R,j} + T^3 r_{V,j}$ be the committed part of each slot and $\hat{\ell}(T) = (\hat{r}_{1:7}(T) \| T^3 \ell_{\hat{V}})$ that of $\ell(T)$, and let $\bar{f}(T) = \hat{f}(T) - \langle c(T), \hat{\ell}(T) \rangle + \hat{r}_0(T)$ be the residual polynomial, the part of f that involves neither r_S nor ℓ_S and is therefore fixed by the time the prover forms C_S. Slots $1, \dots, 7$ then cancel one power each diagonally ($r_{S,j}$ enters slot j at T^{-1}, meeting βT^{a_j} at $T^{a_j - 1}$), while the constant entries of $c_{\ell,V}$ pair the T^{-1} masks into the T^{-1} row, so that slot 0, which enters through $-r_0(T)$ with divisor -1, is solved last:

$$\begin{aligned} r_{S,j} &= \frac{\bar{f}[T^{p_j}]}{\beta}, \qquad (p_1, \dots, p_7) = (-2, 0, 1, 2, 4, 5, 6), \\ r_{S,0} &= -\Big(\bar{f}[T^{-1}] - \sum_{m=1}^{6} \lambda^{g_m} r_{S,m+1} - \lambda^{g_7} \ell_S \Big). \end{aligned} \tag{31}$$

The system is always solvable, $\beta \neq 0$ being sufficient, and with this choice $f(T) \equiv 0$ as (28) requires: the eight error rows cancel by construction, the value row by (29), and every other coefficient is a structural zero. This is the completeness of the protocol.

5.2 Zero-knowledge

Theorem 1 (Zero-knowledge, batched)**.** *For every (M, b), the batched exact-form reciprocal range proof is perfect honest-verifier zero-knowledge outside a challenge set of density $O(1)/q$. After Fiat–Shamir it is zero-knowledge in the random-oracle model.*

Simulator. The transcript to be simulated is $\big((C_O, C_L, C_R),\ C_S,\ \Pi(\tau)\big)$; the fold messages are deterministic functions of $\Pi(\tau)$ and the challenges. The simulator takes only public data:

Simulator $\mathcal{S}$ (input: $V_1, \dots, V_M$ and the challenges).

1. Set $\mu = \rho^2$; form $p_n(\tau)$, $p_s(\tau)$, $c(\tau)$, $P(\tau)$ from (23)–(24) and $\hat{V} = 2 \sum_i \lambda^{i-1} V_i$.
2. Sample $C_L, C_O, C_R \xleftarrow{\$} \mathbb{G}$.
3. Sample $n(\tau) \xleftarrow{\$} \mathbb{F}_q^{n_m}$, $\ell(\tau) \xleftarrow{\$} \mathbb{F}_q^8$; set $\sigma(\tau) = \langle c(\tau), \ell(\tau) \rangle + \|n(\tau)\|_\mu^2$.
4. Solve C_S from the opening identity of (25):
$$C_S = \tau\big(\sigma(\tau) G + \langle \ell(\tau), \mathbf{H} \rangle + \langle n(\tau), \mathbf{G} \rangle - P(\tau) - \delta C_O - \tau C_L - \tau^2 C_R - \tau^3 \hat{V}\big).$$
5. Run the honest norm-linear prover on $(\sigma(\tau), \ell(\tau), n(\tau))$ to emit the ν round pairs (X_i, R_i) and the base-case scalars.

The norm-linear argument provides no hiding and requires none: every message after τ is a deterministic function of $\Pi(\tau) = (\ell(\tau), n(\tau))$ and the challenges, and $\sigma(\tau)$ is determined by $\Pi(\tau)$.

All blinding occurs before the folds. It therefore suffices to prove that in a real transcript (i) the $8 + n_m$ coordinates of $\Pi(\tau)$ are independently uniform, and (ii) (C_O, C_L, C_R) is uniform in $\mathbb{G}^3$ conditioned on $\Pi(\tau)$; C_S is then determined in either case, and by completeness the real C_S satisfies the same equation Step 4 solves. Write the fresh blinding as $\omega = (n_S,\ \ell_S,\ r_O,\ r_L,\ r_R)$: with the zero pattern of (22), $n_m + 16$ free coordinates. After r_S is solved by (31), every coordinate of $\Pi(\tau)$ is a polynomial in ω (not affine: the solved r_S carries $\|n_S\|_\mu^2$ and cross terms $\langle n_S, \cdot\rangle_\mu$ through the $\hat{f}$ coefficients).

(i) Pivots for the opening. The change of variables factorizes into a layer that carries all the parameter dependence diagonally and a fixed 15-dimensional layer that is identical at every (M, b).

Mask layer. By (26), for every k,

$$n(\tau)_k = \tau^{-1} n_{S,k} + (\text{terms without masks}), \qquad \ell(\tau)_8 = \tau^{-1}\ell_S,$$

each mask coordinate appearing with coefficient τ^{-1} in its own row and in no other coordinate of $\Pi(\tau)$ except the rows $\ell(\tau)_{1:4}$ (through the mask cross-terms of $\hat{f}[T^{-2}], \hat{f}[T^0], \hat{f}[T^1], \hat{f}[T^2]$ inside the solved r_S) and the scalar slot $r_{S,0}$, which enters only $\sigma(\tau)$. The masks therefore form a diagonal block of size $n_m + 1$, in which the norm length enters only as the block size.

Committed layer. The remaining 15 blinding coordinates map to the seven rows $\ell(\tau)_{1:7}$ and eight leftovers. Their complete occurrence list in $\Pi(\tau)$ (coefficients in parentheses; entries via the solved r_S carry $1/(\beta\tau)$- or $1/\tau$-type factors):

component	occurrences
$r_{O,0}$	$\ell_2\ (\delta/(\beta\tau))$
$r_{O,1}$	$\ell_1\ (\delta)$
$r_{O,2}$	$\ell_2\ (\delta - \delta\lambda^{g_1}/(\beta\tau)),\quad \ell_3\ (-\delta/\tau)$
$r_{O,3}$	$\ell_2\ (-\delta\lambda^{g_2}/(\beta\tau)),\quad \ell_3\ (\delta),\quad \ell_4\ (-\delta/\tau)$
$r_{O,5}$	$\ell_2\ (-\delta\lambda^{g_4}/(\beta\tau)),\quad \ell_5\ (\delta),\quad \ell_6\ (-\delta/\tau)$
$r_{O,6}$	$\ell_2\ (-\delta\lambda^{g_5}/(\beta\tau)),\quad \ell_6\ (\delta),\quad \ell_7\ (-\delta/\tau)$
$r_{L,0}$	$\ell_3\ (1/(\beta\tau))$
$r_{L,1}$	$\ell_1\ (\tau),\quad \ell_2\ (-1/\tau)$
$r_{L,2}$	$\ell_2\ (\tau),\quad \ell_3\ (-\lambda^{g_1}/(\beta\tau)),\quad \ell_4\ (-1/\tau)$
$r_{L,4}$	$\ell_3\ (-\lambda^{g_3}/(\beta\tau)),\quad \ell_4\ (\tau),\quad \ell_5\ (-1/\tau)$
$r_{L,5}$	$\ell_3\ (-\lambda^{g_4}/(\beta\tau)),\quad \ell_5\ (\tau),\quad \ell_7\ (-1/\tau)$
$r_{R,0}$	$\ell_4\ (1/(\beta\tau))$
$r_{R,1}$	$\ell_1\ (\tau^2),\quad \ell_3\ (-1/\tau)$
$r_{R,3}$	$\ell_3\ (\tau^2),\quad \ell_4\ (-\lambda^{g_2}/(\beta\tau)),\quad \ell_5\ (-1/\tau)$
$r_{R,4}$	$\ell_4\ (\tau^2 - \lambda^{g_3}/(\beta\tau)),\quad \ell_6\ (-1/\tau)$
ℓ_S	$\ell_8\ (\tau^{-1})$
$n_{S,k}$	$n_k\ (\tau^{-1});\quad \ell_1, \ldots, \ell_4$ (mask cross-terms in $\hat{f}[T^{-2}], \hat{f}[T^0], \hat{f}[T^1], \hat{f}[T^2]$)

The parameters enter only through the shape constants λ^{g_m}; the list is otherwise independent of

(M, b). Order the rows $\ell(\tau)_1, \ldots, \ell(\tau)_8, n(\tau)_0, \ldots, n(\tau)_{n_m-1}$. A *pivot* for a row is a component of ω in which it is affine with nonzero coefficient and which enters no later row:

coordinate	**pivot**	**coeff.**	**through**
$\ell(\tau)_1$	$r_{O,1}$	δ	direct
$\ell(\tau)_2$	$r_{O,0}$	$\delta/(\beta\tau)$	$r_{S,2}$, $\bar{f}[T^0]$
$\ell(\tau)_3$	$r_{L,0}$	$1/(\beta\tau)$	$r_{S,3}$, $\bar{f}[T^1]$
$\ell(\tau)_4$	$r_{R,0}$	$1/(\beta\tau)$	$r_{S,4}$, $\bar{f}[T^2]$
$\ell(\tau)_5$	$r_{L,4}$	$-1/\tau$	$r_{S,5}$, $\bar{f}[T^4]$
$\ell(\tau)_6$	$r_{R,4}$	$-1/\tau$	$r_{S,6}$, $\bar{f}[T^5]$
$\ell(\tau)_7$	$r_{L,5}$	$-1/\tau$	$r_{S,7}$, $\bar{f}[T^6]$
$\ell(\tau)_8$	ℓ_S	τ^{-1}	scalar mask
$n(\tau)_0, \ldots, n(\tau)_{n_m-1}$	$n_{S,0}, \ldots, n_{S,n_m-1}$	τ^{-1}	norm mask

The pivot ordering and the coefficient column are the same at every (M, b); only the residuals carry λ^{g_m}. Triangularity is read off the occurrence list: the slot-0 components $r_{O,0}, r_{L,0}, r_{R,0}$ and $r_{O,1}$ occur in exactly one row each; $r_{L,4}, r_{R,4}, r_{L,5}$ occur additionally only in ℓ_3, ℓ_4, ℓ_5, rows that precede the ones they pivot; and the masks occur in no later row, their remaining appearances reaching only $r_{S,0}$, which enters $\sigma(\tau)$ and is itself determined by $\Pi(\tau)$. Solving from the bottom row upward, the $8 + n_m$ coordinates of $\Pi(\tau)$ are independently uniform, which proves (i). The value row stays free of blinding throughout: slot 1 of $c_{\ell,V}$ is zero, $c(T)$ has no T^4 monomial, and $r_{O,4} = r_{L,3} = r_{R,2} = 0$.

(ii) Randomizers for C_L, C_O, C_R. No single component of ω can serve as a pivot for a commitment: every leftover blinding component occurs both in its commitment and in some ℓ row, so the triangular table cannot be extended by three commitment rows. Instead, fixing $\Pi(\tau)$ leaves the blinding underdetermined, and we exhibit three re-randomizing *moves*, each parameterized by a free scalar $t \in \mathbb{F}_q$. A move perturbs one leftover coordinate and corrects the resulting change in $\Pi(\tau)$ using components that occur in exactly one opening row $(r_{O,0}, r_{O,1}, r_{L,0})$, so the correction spills nowhere else. The only parameter-dependent entry is the feedback λ^{g_1} in w_O, forced by the occurrence of $r_{O,2}$ in ℓ_2 through the shape block:

move (t free)	**shift in C_O**	**shift in C_L**	**shift in C_R**
$w_O:\ r_{O,2}{+}t, r_{O,0}{-}(\beta\tau{-}\lambda^{g_1})t, r_{L,0}{+}\beta\delta t$	$t\big(H_1 - (\beta\tau{-}\lambda^{g_1})G\big)$	$t\beta\delta G$	0
$w_L:\ r_{L,1}{+}t, r_{O,1}{-}\frac{\tau}{\delta}t, r_{O,0}{+}\frac{\beta}{\delta}t$	$t\big(\frac{\beta}{\delta}G - \frac{\tau}{\delta}H_0\big)$	tH_0	0
$w_R:\ r_{R,1}{+}t, r_{O,1}{-}\frac{\tau^2}{\delta}t, r_{L,0}{+}\beta t$	$-t\frac{\tau^2}{\delta}H_0$	$t\beta G$	tH_0

Invariance of $\Pi(\tau)$ under w_O, directly from the occurrence list ($r_{O,2}$ appears only in ℓ_2, ℓ_3):

$$\ell_2 : \delta\Big(1 - \tfrac{\lambda^{g_1}}{\beta\tau}\Big)t + \tfrac{\delta}{\beta\tau}\big(-(\beta\tau - \lambda^{g_1})t\big) = 0, \qquad \ell_3 : -\tfrac{\delta}{\tau}t + \tfrac{1}{\beta\tau}(\beta\delta t) = 0;$$

w_L (repairing ℓ_1, ℓ_2) and w_R (repairing ℓ_1, ℓ_3) cancel the same way, with no parameter dependence. After a move, r_S is re-solved and C_S recomputed, so the verification identity holds throughout.

Running the moves with independent t_O, t_L, t_R and passing to discrete logs base G ($h_i = \log_G H_i$, $\log_G G = 1$), the commitment shifts form the columns of

$$M_{\mathrm{mv}} = \begin{pmatrix} h_1 - (\beta\tau - \lambda^{g_1}) & \dfrac{\beta - \tau h_0}{\delta} & -\dfrac{\tau^2 h_0}{\delta} \\ \beta\delta & h_0 & \beta \\ 0 & 0 & h_0 \end{pmatrix}, \qquad \det M_{\mathrm{mv}} = h_0\big(h_0 h_1 + \lambda^{g_1} h_0 - \beta^2\big). \tag{32}$$

The determinant is quadratic in β with leading coefficient $-h_0 \neq 0$, so for any nontrivial generators and any λ it vanishes for at most two values of β. Outside this event the moves range over all of $\mathbb{G}^3$ in (C_O, C_L, C_R) while fixing $\Pi(\tau)$, which proves (ii).

Assembly. The mask layer, the triangular layer, and the moves compose into one bijection of the blinding space,

$$\Phi : \ \mathbb{F}_q^{n_m+16} \xrightarrow{\ \sim\ } \underbrace{\mathbb{F}_q^{n_m+8}}_{\Pi(\tau)} \times \underbrace{\mathbb{F}_q^{5}}_{\text{free}} \times \underbrace{\mathbb{F}_q^{3}}_{(C_O, C_L, C_R)}, \tag{33}$$

valid whenever $\tau, \beta, \delta \neq 0$, $h_0 \neq 0$, and $\beta^2 \neq h_0 h_1 + \lambda^{g_1} h_0$, under which the opening and the commitment triple are coordinate projections. Projections of the uniform distribution are uniform and independent, so Steps 2–3 of $\mathcal{S}$ match the real distribution exactly; Steps 1, 4, and 5 compute the same deterministic functions as the real prover. Together with the invertibility requirements the total zero-knowledge defect is $O(1)/q$, uniformly in (M, b). □

5.3 Knowledge soundness

Theorem 2 (Exact-form soundness, batched)**.** *Assume discrete-logarithm hardness in $\mathbb{G}$. For every (M, b), if a prover convinces the verifier with non-negligible probability, then it knows (v_i, s_i) with $V_i = v_i G + s_i H_0$ and $0 \leq v_i < 2^b$ for every $i = 1, \ldots, M$, with no assumption on how the V_i were formed.*

Lemma 1 (Interpolation)**.** *A polynomial over $\mathbb{F}_q$ of degree at most d that vanishes at $d+1$ distinct points is zero; a Laurent polynomial supported on $n+1$ consecutive powers that vanishes at $n+1$ distinct nonzero points is zero (multiply by the power clearing the support).*

Lemma 2 (Reciprocal vanishing)**.** *Let $U \subset \mathbb{F}_q$ be a finite pole set and $e : U \to \mathbb{F}_q$. If $\sum_{p \in U} e_p/(x+p)$ vanishes at $|U|$ distinct points x, none a pole, then every $e_p = 0$.*

Proof. Multiplying by $\prod_{p \in U}(x+p)$ gives a polynomial of degree at most $|U|-1$ vanishing at $|U|$ points, hence zero by Lemma 1; evaluating it at $-p$ leaves $e_p \prod_{p' \neq p}(p' - p) \neq 0$ unless $e_p = 0$. □

The extractor builds a tree of accepting transcripts; to fork a challenge it re-runs the prover with identical prior coins and a fresh value, holding the same-round challenges fixed. We describe the fork levels from the innermost outward. All extracted openings are representations over the key $(G, \mathbf{H}, \mathbf{G})$; at any level, two distinct representations of the same group element yield a discrete-logarithm relation on the key and count as a successful extraction, so below we may treat the opened representations as unique and compute with them as formal identities. The challenges are kept in the good set ($\tau, \rho, \beta, \delta, \alpha$ nonzero and $\alpha \notin \{-j\}_{j \leq 15}$), which excludes $O(1)$ values per fork.

Folding (γ: 4 per round). In one norm-linear round the verifier accepts $C + \gamma X + (\gamma^2{-}1)R = \sigma_i G + \langle \ell_i, \mathbf{H}\rangle + \langle n_i, \mathbf{G}\rangle$. Three distinct γ open C, X, R over the key, since $1, \gamma, \gamma^2{-}1$ are linearly independent as polynomials in γ. A fourth is needed to certify that the openings fold from one parent: for a generator pair H_{2i}/H_{2i+1} the genuine-fold test $\gamma\ell_{i,0} - \ell_{i,1} = 0$ becomes, once the prover may route contributions through R, a cubic

$$r_0\gamma^3 + (x_0 - r_1)\gamma^2 + (c_0 - r_0 - x_1)\gamma + (r_1 - c_1) = 0, \tag{34}$$

whose vanishing at four points forces it to vanish identically, hence $r_0 = 0$ and parent-length openings. A padded slot carries the identity element and a zero constraint entry, so the identity for it reduces to $\ell'_s = c_{2s}$ and needs no extraction.

The forks recover the parent list itself. Index $2s$ of the parent enters slot s of the fold with weight 1 on $\mathbf{H}$ and ρ on $\mathbf{G}$, index $2s{+}1$ with weight γ; iterating, the slot-s *pivot* after r rounds is the original index $s2^r$, carrying the nonzero product of the round base weights $\rho, \mu, \mu^2, \ldots$ The pivot row of the identity returns the child coordinate at s, the sibling row returns the parent's odd coordinate as the pivot component of X and forces that of R to zero, and the scalar follows: σ is quadratic in γ, and its γ^0 and γ^2 parts reassemble $\langle c, \ell\rangle + \|n\|^2_\mu$ at the parent lengths, each round's base weight squaring to its norm weight.

With ν rounds, 4^ν γ-forks yield an opening $(\sigma(\tau), \ell(\tau), n(\tau))$ of $C(\tau)$ satisfying (27).

Evaluation (τ: 13). $C(\tau)$ is linear in $C_S, C_O, C_L, C_R, \hat{V}$ with coefficients $\tau^{-1}, 1, \tau, \tau^2, \tau^3$, so five distinct τ open all five commitments over the key; the openings populate the Laurent vectors (26) and make (27) the statement $f(\tau) = 0$. The extracted openings are arbitrary: the H_7-coordinates ℓ_O, ℓ_L, ℓ_R need not vanish, which adds $\delta\ell_O + T\ell_L + T^2\ell_R$ to slot 8 of $\ell(T)$ and, paired with the constant λ^{g_7}, reaches only T^0, T^1, T^2; and no zero pattern may be assumed when bounding the support of f: the largest power is βT^7 against the T^3 components of the input coordinates, T^{10}; the smallest is T^{-2}; and the constant entries of $c_{\ell,V}$ contribute nothing above T^3. Hence f is supported on the thirteen powers $T^{-2}, \ldots, T^{10}$, and thirteen distinct τ force $f(T) \equiv 0$ by Lemma 1. In particular the value row (29) vanishes, now as an identity in the extracted openings.

Linear and shape rows (λ: $M(n_m{+}8) + 1$). Fork the step-4 challenge λ, holding $\alpha, \rho, \beta, \delta$ fixed. Across branches the openings of C_O, C_L, C_R are fixed, since these elements precede step 4 and are opened in every branch; the opening of C_S may differ per branch, but by the design invariant no component of C_S appears in (29). The inputs need one step more, since no V_i is ever opened and $\hat{V}$ is a different element in each branch. Over any M of the branches the matrix $2(\lambda_r^{i-1})$ is Vandermonde with distinct nodes, hence invertible, and representations combine linearly, so inverting it carries the M opened representations of $\hat{V}$ to one representation of each V_i: these are the coordinates (21), fixed once for all branches and at no cost in forks, the M branches being among those already taken. In every remaining branch the opened representation of $\hat{V}$ either agrees with the λ-combination of these, or two representations of one element yield a dlog relation and the extraction is complete. The vanishing of (29) is therefore a polynomial identity in λ of degree

$M(n_m{+}8)$ with branch-independent coefficients, one monomial per linear row:

$$\begin{aligned}
\lambda^0 &: \; L_1 + E_0, \\
\lambda^{i-1} &: \; L_i, \qquad i = 2, \ldots, M, \\
\lambda^M &: \; S, \\
\lambda^{M(k+1)+1+(i-1)} &: \; -4n_{V_i,k}, \qquad k = 0, \ldots, n_m{-}1, \quad i = 1, \ldots, M, \\
\lambda^{g_m+(i-1)} &: \; -2h_{i,m}, \qquad m = 1, \ldots, 7, \quad i = 1, \ldots, M.
\end{aligned} \tag{35}$$

$M(n_m{+}8)+1$ distinct λ force every coefficient to zero (Lemma 1; the factors 2 and 4 are invertible by field domination). This establishes all $M(n_m{+}7)$ shape rows,

$$n_{V_i} = 0, \qquad h_{i,m} = 0 \;\; (m = 1, \ldots, 7), \qquad i = 1, \ldots, M,$$

so every V_i lies in the (G, H_0)-plane; the set-membership row $S = 0$; and, since the δ-term of the value link in (30) dies with $n_{V_i} = 0$, the value links for $i \geq 2$,

$$u_i = \sum_{t=0}^{D-1} 16^t n_{L,D(i-1)+t} \qquad (i = 2, \ldots, M).$$

The degree-0 residue $L_1 + E_0 = 0$ passes to the next levels. The distinct value-link powers are what bind the batch: exchanging two values' digit blocks preserves the digit multiset, hence the shared set-membership row, and is caught only by the per-value λ-monomials.

Blinding bracket (β: 2). In the residue $L_1 + E_0 = 0$, only the bracket $\beta(\delta r_{O,4} + r_{L,3} + r_{R,2})$ of (30) depends on β, with branch-independent cofactor; the δ-term of L_1 has already vanished with $n_{V_1} = 0$. Two forks of β remove the bracket, leaving, for every retained (α, ρ),

$$u_1 - \sum_{t=0}^{D-1} 16^t n_{L,t} = \sum_{k=0}^{N_d-1} \mu^{k+1}\big(n_{L,k}n_{R,k} + \alpha n_{R,k} - 1\big) + \sum_{k=N_d}^{n_m-1} \mu^{k+1} n_{L,k}n_{R,k}. \tag{36}$$

Hadamard rows and the first value link (ρ: $n_m{+}1$). Equation (36) is a polynomial identity in μ of degree n_m whose μ^0 coefficient is the first value link and whose μ^{k+1} coefficient is the k-th Hadamard row. Fork ρ through $n_m + 1$ values with distinct squares $\mu = \rho^2$: every coefficient vanishes, giving

$$u_1 = \sum_{t=0}^{D-1} 16^t n_{L,t}, \qquad (\alpha + n_{L,k})n_{R,k} = 1 \;\; (k < N_d), \qquad n_{L,k}n_{R,k} = 0 \;\; (k \geq N_d),$$

the middle family holding for each retained value of α.

Digits (α: $N_d{+}16$). By the Hadamard rows, $n_{R,k} = (\alpha{+}n_{L,k})^{-1}$ at every retained α (in particular $\alpha + n_{L,k} \neq 0$). Substituting into the set-membership row $S = 0$ of (30) and partial-fractioning $j/(\alpha(\alpha{+}j)) = 1/\alpha - 1/(\alpha{+}j)$ turns it into a vanishing sum of simple poles in α:

$$\begin{aligned}
&\sum_{q \in U} \frac{e_q}{\alpha + q} = 0, \qquad U = \{n_{L,k}\}_{k<N_d} \cup \{0, 1, \ldots, 15\}, \qquad |U| \leq N_d + 16, \\
&e_q = \big[q{=}0\big]\Big(N_d - \sum_{j=1}^{15} n_{O,j-1}\Big) - \big|\{k < N_d : \; n_{L,k} = q\}\big| + \sum_{j=1}^{15} \big[q{=}j\big] n_{O,j-1}.
\end{aligned} \tag{37}$$

$N_d + 16$ retained values of α, none a pole, force every residue e_q to zero by Lemma 2. If some digit-slot value n_{L,k_0} lay outside the table $\{0, \ldots, 15\}$, its residue would be the negative of its multiplicity, a nonzero integer at most N_d and so nonzero in $\mathbb{F}_q$ by field domination, contradicting $e_q = 0$. Hence every $n_{L,k} \in \{0, \ldots, 15\}$ for $k < N_d$ (and, as a by-product, the residues at the table points identify $n_{O,j-1}$ with the shared multiplicities).

Output. The extractor reads the digits off the **G**-representation of C_L and the blindings off the H_0-coordinates of the inputs:

$$\begin{aligned} d_{i,t} &:= n_{L,D(i-1)+t}, \qquad v_i := \sum_{t=0}^{D-1} 16^t d_{i,t} \ \in \ [0, 2^b), \qquad s_i := h_{i,0}, \\ V_i &= u_i G + h_{i,0} H_0 = v_i G + s_i H_0, \qquad i = 1, \ldots, M. \end{aligned} \tag{38}$$

The range bound holds by construction, every digit being a table entry. The representation collapses by the shape rows together with the value links, $2^b \ll q$ identifying the field element u_i with the natural number v_i, so the exact opening is a consequence of the extraction rather than an assumption on the form of the V_i.

Challenge	**Outcome**	**#**
γ (per NL round, ν rounds)	invert one fold	4
τ	$f(T) \equiv 0$; open $C_S, C_O, C_L, C_R, \hat{V}$	13
λ	shape rows; $S = 0$; value links $i \geq 2$	$M(n_m{+}8) + 1$
β	blinding bracket removed	2
ρ ($\mu = \rho^2$ distinct)	value link 1; Hadamard rows	$n_m + 1$
α	every digit in $\{0, \ldots, 15\}$	$N_d + 16$

Table 1: Fork counts for the batched extractor.

For fixed parameters all branch counts in Table 1 are constants, so the extractor runs in expected polynomial time, and under discrete-logarithm hardness a convincing prover knows valid (v_i, s_i) for every i. Fiat–Shamir preserves knowledge soundness in the random-oracle model at the same challenge points. □

6 Evaluation

We implemented the protocol from scratch in Rust over the Ristretto255 group (`curve25519-dalek` v4)[3]. All measurements were taken on an Apple M2 Max on a single thread. Table 2 compares proving and verification times with Bulletproofs (BP), using the `bulletproofs` crate v5.0.0 with generators built once per configuration as the baseline, for a range of configurations. Across them, proving is between 1.7× and 7.8× and verification between 1.1× and 4.5× faster. A single 64-bit proof takes about 1.3 ms to produce and about 0.32 ms to verify, and occupies 416 bytes ($10\mathbb{G} + 3\mathbb{F}$) against 672 bytes ($16\mathbb{G} + 5\mathbb{F}$) for Bulletproofs, a reduction of 38%. In general a proof for M values of b bits has $64 \log_2 n_m + 160$ bytes, where n_m is $\max(Mb/4, 16)$ rounded up to a power of two. The verifier's multi-scalar multiplication (MSM) runs over 36 bases (G, **H**, **G**, the five commitments and the six round messages) against 147 for Bulletproofs.

[3] `https://github.com/MystenLabs/fastcrypto/tree/main/fastcrypto/src/bulletproofspp`

	Proving			Verification		
Config	BP	BP++	Speedup	BP	BP++	Speedup
16-bit × 1	2.10	1.24	1.7×	0.36	0.31	1.1×
32-bit × 1	3.86	1.26	3.1×	0.57	0.31	1.8×
64-bit × 1	7.28	1.29	5.6×	1.00	0.32	3.2×
16-bit × 4	7.89	1.39	5.7×	1.06	0.35	3.0×
16-bit × 8	15.40	2.08	7.4×	1.74	0.49	3.6×
32-bit × 8	28.42	3.97	7.2×	2.85	0.68	4.2×
64-bit × 16	106.5	13.61	7.8×	8.53	1.92	4.4×
64-bit × 32	205.1	27.11	7.6×	15.94	3.55	4.5×

Table 2: Proving and verification time in ms of Bulletproofs (BP) and BP++ for base 16 over Ristretto255 on an Apple M2 Max.

Optimizations. The reported numbers rely on the following optimizations of the construction of Section 4.

- **Lazy base folding (NLA prover).** In each round of the NLA the prover folds the bases $\mathbf{G}$ and $\mathbf{H}$ and computes the commitments X and R, which together dominate its cost. Instead of folding the bases in every round, we compute each round's X and R with a single MSM over the round's implicit bases and fold the bases only every B rounds. This is correct because folding is linear: k rounds after the last fold, every implicit base element is a fixed 2^k-term combination of the last folded bases, and substituting these combinations into X and R yields the same group elements and hence bit-identical proofs. It is faster because a scalar multiplication is dominated by a ladder of about 256 point doublings, which an MSM pays once for all of its terms. Per folded base element, direct point arithmetic pays two ladders (≈56 μs), a per-round two-term MSM pays one (≈28 μs), and a fold batched over $B = 3$ rounds pays one eighth (≈7.3 μs). We determined $B = 3$ empirically; $B = 1$ recovers per-round folding.

- **No combined commitment on the prover side (NLA prover).** The prover never needs the group element $C(\tau)$: it writes only the round commitments X, R to the transcript, and the NLA operates on the opening of $C(\tau)$. We therefore skip its computation and the per-round folding of the commitment, which removes several MSMs and scalar multiplications per proof.

- **Fixed-base precomputation with a size-dependent fallback (prover and verifier).** The generators $G, \mathbf{H}, \mathbf{G}$ are system parameters fixed across all proofs. We precompute tables of odd multiples for them once (Dalek's precomputed Straus algorithm) and reuse them in every MSM whose points are generators: the four commitments C_L, C_O, C_R, C_S, the commitments X, R of all NLA rounds before the first fold, and the verifier's final check. Reading a generator's multiples from a table instead of recomputing them lowers the MSM cost by ≈1.2× at 265 points. The advantage does not persist at scale: Straus-type algorithms, precomputed or not, are overtaken by Pippenger's bucket method as the MSM grows, and Dalek's precomputed interface offers no Pippenger variant. Each call therefore chooses between the two by a weighted point count, using the tables while $n_{\text{static}} + 3n_{\text{dynamic}} \leq 600$, where the dynamic points are those without tables. The rule was calibrated on the M2 Max: the crossover lies

at ≈590 static points without dynamic terms, at ≈515 with 25 and at ≈468 with 55 dynamic terms; near the crossover the two methods differ by at most 2%, so misplacing the threshold by 50 points costs less than 1%.

- **CRS caching (prover and verifier).** The generators and their tables depend only on the length of the norm vector n, so we derive them once per length, one hash-to-curve per generator plus the table build, and cache them for the lifetime of the process. All reported timings are steady-state against this cache; re-deriving the CRS on every call would add several hundred hash-to-curve operations per proof at the largest configurations.

- **Single-MSM verification with a challenge-weight table (NLA verifier).** The verifier collapses its entire check into one MSM over the fixed bases $G, \mathbf{H}, \mathbf{G}$, the commitments C_L, C_O, C_R, C_S, the round commitments X_r, R_r and the input commitment V, and tests the result against the identity element. The combined commitment $C(\tau)$ is never materialised: its public coefficients over $\mathbf{G}$ are subtracted inside the base weights, and its remaining terms enter the same MSM directly. The weight of each element of $\mathbf{G}$ is the product of one final-witness scalar and κ round factors, where κ is the number of folding rounds and round r contributes either the challenge γ_r or the weight ρ_r, selected by one bit of the element's index. We precompute all 2^κ products of round factors in a table: starting from $W_0 = (1)$, round r extends it as $W_r = (\rho_r W_{r-1} \| \gamma_r W_{r-1})$ with entry-wise multiplication, so that W_κ holds every product at one field multiplication per entry. Two such tables are kept, matching the two folding rules: one for $\mathbf{G}$ from the pairs (ρ_r, γ_r) and one for $\mathbf{H}$ from $(1, \gamma_r)$. Each base weight then costs one lookup and one field multiplication instead of a κ-fold product, which reduces the weight computation from $O(\mathrm{len}(n)\kappa) = O(\mathrm{len}(n)\log\mathrm{len}(n))$ to $O(2^\kappa + \mathrm{len}(n)) = O(\mathrm{len}(n))$ field multiplications, since $2^\kappa \le \mathrm{len}(n)$. This adapts the scalar-computation optimization of Bulletproofs [1] to the two-scalar folding of the NLA.

- **Pre-weighting and cancellation of weighted inner products (NLA prover and verifier).** The scalar work of both parties is dominated by weighted inner products $\langle a, b\rangle_\mu = \sum_i a_i b_i \mu^{i+1}$, and two observations reduce it: all of them share the weight μ, and the largest block of them cancels structurally. First, evaluated directly, a weighted inner product costs three field multiplications per element, two for $a_i b_i \mu^{i+1}$ and one to advance the power of μ. Writing $\langle a, b\rangle_\mu = \langle a \circ \bar{\mu}, b\rangle$ with $\bar{\mu} = (\mu, \mu^2, \ldots)$, we weight each recurring vector by $\bar{\mu}$ once, after which every weighted inner product it participates in is a plain inner product at one multiplication per element. Second, we avoid the products that cancel. The prover needs the coefficients of the error polynomial $\hat{f}(T) = p_s(T) + \hat{v}T^3 - \|n(T)\|_\mu^2$, where $n(T) = w(T) + p_n(T)$ splits into the committed part $w(T) = T^{-1}n_S + \delta n_O + T n_L + T^2 n_R$ (Section 4.7; $n_{\hat{V}} = 0$ for the honest prover) and the public part $p_n(T)$. The public square $\|p_n(T)\|_\mu^2$ inside $p_s(T)$ cancels against the expansion of $\|n(T)\|_\mu^2$, since $\|n(T)\|_\mu^2 - \|p_n(T)\|_\mu^2 = \langle w(T), n(T) + p_n(T)\rangle_\mu$, leaving

$$\hat{f}(T) = \big(\hat{v} + 2(\lambda_{a_l} + \mu_{a_m})\big)T^3 - \langle w(T), n(T) + p_n(T)\rangle_\mu.$$

  Only the $4 \times 5 = 20$ bilinear products between the four vector coefficients of w and the five of $n + p_n$ are evaluated; the public-public products are never computed. The verifier needs no coefficient of $p_s(T)$ at all, since it evaluates $p_s(\tau) = \|p_n(\tau)\|_\mu^2 + 2(\lambda_{a_l} + \mu_{a_m})\tau^3$ directly, one weighted norm of the vector $p_n(\tau)$ that its single-MSM check requires anyway.

- **Sparse λ-power families (prover and verifier).** The constraint vectors of the batched construction read exactly two families of powers of λ: the per-value weights $\lambda^0, \ldots, \lambda^M$ and the shape-row powers λ^{Mj+1} for $j = 1, \ldots, \mathrm{len}(n) + 7$. We build these two families directly,

the second by stepping with λ^M, instead of the dense vector of powers up to $\lambda^{M(\mathrm{len}(n)+8)}$. Of its $M(\mathrm{len}(n)+8)+1$ entries, the $(M-1)(\mathrm{len}(n)+7)$ interior powers of the per-coordinate bands are never read, since only the soundness analysis separates them, and this fraction grows with the batch size.

- **Variable-time Pedersen commitments (prover).** We compute each input commitment $V = vG + sH_0$ as one two-term variable-time MSM instead of two constant-time scalar multiplications and an addition (≈31 μs against ≈56 μs per commitment). The gain has two sources: the joint computation shares the ≈256-doubling ladder between both terms, which separate multiplications pay twice, and the variable-time path recodes the scalars into non-adjacent form, whose sparse signed digits require point additions only at the ≈1/6 of positions with nonzero digits, whereas the constant-time path processes every window uniformly.

7 Errata relative to the original paper

Two formulas in [BP++ §5.2.3–5.2.4] are inconsistent with the equations surrounding them: following either as printed, the honest prover produces proofs the verifier rejects. Both were isolated by a sign search on a minimal instance with $\mathrm{len}(n) = 2$ and confirmed on the full circuit with $\mathrm{len}(n) = 16$.

1. **The affine T^3 term of $p_s(T)$ lacks a factor of 2** [BP++ Eq. (46)].

   The paper defines
   $$p_s(T) = \|p_n(T)\|_\mu^2 + \langle \boldsymbol{\lambda}, a_l\rangle T^3 + \langle \boldsymbol{\mu}, a_m\rangle T^3,$$
   with the affine terms at coefficient 1. Every witness contribution to the value coefficient $\hat{f}[T^3]$ of [BP++ Eq. (49)] is doubled: the linear constraint vector $\hat{c}_l(T)$ carries an explicit factor of 2 [BP++ Eq. (48)], the aggregated input $\hat{V}$ carries a factor of 2 [BP++ Eq. (44)], and the norm expansion doubles the cross-terms $2\langle p_n, \hat{n}\rangle_\mu$ and $2\langle n_L, n_R\rangle_\mu$. The circuit identity [BP++ Eq. (34)] balances these witness rows against the affine constants at coefficient one, so the affine terms must be doubled with them:
   $$p_s(T) = \|p_n(T)\|_\mu^2 + 2\big(\langle \boldsymbol{\lambda}, a_l\rangle + \langle \boldsymbol{\mu}, a_m\rangle\big)T^3,$$
   as instantiated in (15). With the printed coefficient, $\hat{f}[T^3]$ differs from zero by the affine sum on honest witnesses, which is nonzero for all but $O(1)$ challenges.

2. **The slot-0 component of r_S carries the wrong sign** [BP++ §5.2.4, Eqs. (58)–(59)].

   The blinding derivation rewrites the verification polynomial [BP++ Eq. (35)] as $f(T) = \hat{f}(T) - \langle \hat{c}_r(T), r(T)\rangle$ [BP++ Eq. (58)], with $\hat{c}_r(T) = (1, \beta T^{-1}, \beta T, \beta T^2, \beta T^3, \beta T^5, \beta T^6, \beta T^7)$, and solves $f \equiv 0$ by the substitution $r_S = (\hat{f}_0 \| \beta^{-1}\hat{f}_{1:}) - s_r$ [BP++ Eq. (59)]. The rewriting flips the sign of the G-side blinding: $r_{X,0}$ is the G-coordinate of C_X [BP++ Eq. (36)], so the combined commitment opens with $+r_0(T)$ on G [BP++ Eq. (51)], and [BP++ Eq. (35)] gives
   $$f(T) = \hat{f}(T) + r_0(T) - \big\langle \hat{c}_{r,1:}(T), r_{1:}(T)\big\rangle,$$
   in which slot 0 acts with -1 where [BP++ Eq. (58)] has $+1$. The substitution must therefore negate its first component,
   $$r_S = \big(-\hat{f}_0 \big\| \beta^{-1}\hat{f}_{1:}\big) - s_r,$$
   which is the factor -1 of (18); the β-asymmetry of [BP++ Eq. (59)], slot 0 undivided by β, is correct as printed. With the printed sign, the T^{-1} coefficient of f fails to cancel on honest witnesses.

Acknowledgements

We thank Ben Riva and Jonas Lindstrøm for their insightful feedback on earlier drafts of this paper and for their thorough review of the source code.

References

[1] Benedikt Bünz, Jonathan Bootle, Dan Boneh, Andrew Poelstra, Pieter Wuille, and Greg Maxwell. *Bulletproofs: Short Proofs for Confidential Transactions and More.* IEEE Symposium on Security and Privacy, 2018. `https://eprint.iacr.org/2017/1066`.

[2] Liam Eagen, Sanket Kanjalkar, Tim Ruffing, and Jonas Nick. *Bulletproofs++: Next Generation Confidential Transactions via Reciprocal Set Membership Arguments.* Cryptology ePrint Archive, Report 2022/510, 2022. `https://eprint.iacr.org/2022/510`.