Primal-Shortened ISD Key Recovery on Binary Goppa Codes

MiniMax

November 2026

Abstract

Binary Goppa codes form the algebraic foundation of the McEliece cryptosystem and the NIST post-quantum candidate Classic McEliece. To evaluate concrete key-recovery security margins across code parameters, the Technology Innovation Institute (TII) McEliece Challenges provide 206 structural key-recovery instances (Track 2). Existing algebraic key-recovery attacks rely either on shortened-dual rank-2 and higher-order vanishing (HOV/HOVER) systems, whose standard kernels collapse at moderate code rates, or on primal hold-out attacks, whose high-order derivative systems still span up to tens of millions of columns after shortening.

We present a primal-shortened key-recovery framework (Route A) built on a direct structural observation: although the public parity-check matrix scrambles the rows of the private Goppa matrix, it leaves the primal codeword space invariant, where every minimum-weight codeword obeys an exact quadratic identity over the secret support. Restricting the primal code to a compact column window before sampling compresses this degree-2 interpolation system by up to 6.59× without penalizing multi-target Information-Set Decoding (ISD) yield. Pairing a cache-resident multi-target ISD engine with a systematic $\mathbb{F}_2$-linear lifting solver reduces key recovery to lightweight codeword harvesting plus seconds of dense binary linear algebra, complemented by an accelerated HOVER solver (Route B) on one high-rate parameter family.

Across the official TII catalog, our framework recovers 94 Track-2 private keys (89 via Route A and 5 via Route B), achieving 55 first-blood recoveries and accelerating 39 previously solved instances, with all 94 keys verified by TII's official SageMath checker. Compared to the shortened-dual records on `TII-166` and `TII-125` (9.20 h and 9.47 h on 72 cores, CRYPTO 2026), Route A finishes in 5.36 s and 5.90 s on a single 64-core CPU server, achieving 6,968× and 6,493× core-hour speedups. On `TII-254` and `TII-253`, it finishes in 30.4 s and 1.80 s, achieving 1,896× and 43,400× wall-time speedups over the prior GPU-cluster record and the multi-core hold-out record by Anthropic. Finally, our codeword-supply analysis shows that Classic McEliece standard parameters lie far outside the feasible regime of this attack.

1 Introduction

Code-based public-key cryptography, initiated by McEliece [28] in 1978 and reformulated in dual syndrome form by Niederreiter [29], is one of the oldest approaches to public-key encryption that resists quantum attacks. Its security rests on two distinct assumptions: the average-case hardness of bounded-distance syndrome decoding in random binary linear codes (whose worst-case general decoding problem is NP-complete [4]), and the structural hardness of distinguishing a public Goppa parity-check matrix $H_{\text{pub}} \in \mathbb{F}_2^{mt \times n}$ from a uniform matrix or recovering its secret algebraic representation. Whereas high-rate alternant instances and compact algebraic proposals such as generalized Reed–Solomon subcodes succumbed to structural distinguishers and filtration attacks [15, 10], binary Goppa codes [31, 25] have withstood nearly five decades of cryptanalysis and form the basis of the NIST post-quantum key-encapsulation candidate Classic McEliece.

To benchmark the concrete complexity of code-based cryptanalysis across code lengths n, extension degrees $m = [\mathbb{F}_{2^m} : \mathbb{F}_2]$, and Goppa degrees t, Kirshanova and Esser [21] introduced the TII McEliece Challenges, organized into three tracks:[1]

1. **Theoretical Key-Recovery Algorithms (Track 1):** Design classical or quantum key-recovery and distinguishing algorithms with improved asymptotic or concrete complexity in secure McEliece parameter regimes. Because this track evaluates algorithmic designs rather than fixed public keys, no challenge files are generated for Track 1 in the companion repository [21].

2. **Practical Structural Key Recovery (Track 2):** Given *only* a concrete public parity-check matrix $H_{\text{pub}} \in \mathbb{F}_2^{mt \times n}$ and the field polynomial $f(x)$ defining $\mathbb{F}_q = \mathbb{F}_2[x]/(f(x))$ $(q = 2^m)$, recover a support tuple $\hat{L} = (\hat{z}_0, \ldots, \hat{z}_{n-1}) \in \mathbb{F}_q^n$ of n distinct elements and a monic square-free degree-t Goppa polynomial $\hat{g}(X) \in \mathbb{F}_q[X]$ $(\hat{g}(\hat{z}_j) \neq 0$; chosen irreducible in TII's generator [21]) whose vertical $\mathbb{F}_2$-expansion $H_{\text{raw}}(\hat{L}, \hat{g}) \in \mathbb{F}_2^{mt \times n}$ of the alternant matrix $\big(\hat{z}_j^i / \hat{g}(\hat{z}_j)\big)_{0 \le i < t,\, 0 \le j < n}$ reproduces the public key in reduced row-echelon form:

$$\mathrm{RREF}\big(H_{\text{raw}}(\hat{L}, \hat{g})\big) = H_{\text{pub}}. \tag{1}$$

3. **Practical Syndrome Decoding (Track 3):** Given $H_{\text{pub}} \in \mathbb{F}_2^{mt \times n}$ and a target syndrome $s = H_{\text{pub}} e^\top \in \mathbb{F}_2^{mt}$ for an unknown weight-t error vector $e \in \mathbb{F}_2^n$, recover e. Treating H_{pub} as a generic binary linear code, this problem is the domain of single-target *Information-Set Decoding (*ISD*)* and code-sieving algorithms, spanning Lee–Brickell [23], Stern–Dumer [37, 14], Canteaut–Chabaud [8], Ball-Collision [6], multi-layer representation (BJMM) and nearest-neighbor decoders [26, 3, 27], and recent advances in code sieving, dual reductions, and concrete cost modeling [13, 9, 5].

Throughout this paper, we study the 206 concrete key-recovery instances of Track 2.

1.1 Prior Key-Recovery Paradigms and Bottlenecks

The 206 Track-2 challenges span extension degrees $m \in \{5, \ldots, 10\}$, Goppa degrees $t \in \{2, \ldots, 24\}$, and security ratings $\lambda_{\text{TII}} \in [22, 255]$. The 36 instances with $\lambda_{\text{TII}} \le 82$ were solved during the initial competition without published keys, while 41 of the 170 instances with $\lambda_{\text{TII}} \ge 83$ were subsequently solved across eight public papers and repositories [30, 2, 7, 18, 38, 33, 34, 35, 39]. Aside from Panny's support-splitting search on small parameters [30] (such as 161 core-days on `TII-91`), published algebraic key-recovery attacks fall into two paradigms.

Shortened-Dual Rank-2 and Higher-Order Vanishing Attacks. Because the rows of $H_{\text{raw}}(L, g)$ are $\mathbb{F}_2$-bit slices of rational evaluations $z_j^i / g(z_j)$ over $\mathbb{F}_{2^m}$, componentwise products of dual codewords satisfy low-rank polynomial relations that do not hold for random linear codes. Building on high-rate alternant distinguishers [15, 10], quadratic-form spaces [11], the syzygy distinguisher [32], and the higher-order vanishing (HOV) distinguisher [19], recent attacks first shorten the dual code $\Gamma(L, g)^\perp$ at s coordinates to compress its dimension before extracting algebraic structure. Bardet, Lemoine, and Tillich [2] (via Pfaffian rank-2 ideals) and Briaud et al. [7] (via bilinear rank-2 modeling) isolate rank-2 forms in the shortened dual square, whereas Hemmert [18] (via low-rank catalecticant MinRank search) and Saarinen [33] (HOVER, via the public $\mathbb{F}_{2^m}$-endomorphism algebra) exploit degree-3 and degree-4 forms that vanish together with all first partial derivatives on the shortened dual code. Although these dual methods account for 37 of the 41 prior Track-2 solutions, their non-trivial vanishing kernels exist only at high

[1] We follow the Track 1–3 numbering of the challenge repository [21], corresponding on the HeroX portal to Track 1A (theoretical key recovery), Track 1B (practical key recovery), and Tracks 2A–2B (message recovery).

Table 1: Flagship key-recovery comparison on TII Track-2 challenges across wall-clock runtime, compute cost, and peak memory.

ID	Parameters	Prior Time	Prior Compute	Our Time	Our Compute	RAM	Speedup / Status
83	$(253, 8, 5, 213)$	24.3 s [35]	0.081 ch (12 cores)	**1.58** s	**0.028** ch	232 MB	15.4× time / 2.9× ch
125	$(507, 9, 7, 444)$	9.47 h [2]	681.8 ch (72 cores)	**5.90** s	**0.105** ch	340 MB	5,778× time / 6,493× ch
166	$(500, 9, 7, 437)$	9.20 h [2]	662.5 ch (72 cores)	**5.36** s	**0.095** ch	340 MB	6,180× time / 6,968× ch
210	$(491, 9, 7, 428)$	4.31 h [2]	310.2 ch (72 cores)	**6.04** s	**0.107** ch	340 MB	2,568× time / 2,899× ch
253	$(214, 8, 9, 142)$	21.7 h [39]	694.4 th (32 threads)	**1.80** s	**0.032** ch	242 MB	43,400× time / 21,688× th
253	$(214, 8, 9, 142)$	14.0 h [38]	1×H200 + 48 threads	**1.80** s	**0.032** ch	242 MB	28,000× time (CPU vs. GPU)
223	$(492, 9, 9, 411)$	47.75 h [18]	12,224 ch (256 cores)	**39.9** s	**0.709** ch	326 MB	4,308× time / 17,241× ch
129	$(509, 9, 9, 428)$	4.51 h [33]	126.2 vch (28 vCPUs)	**41.3** s	**0.734** ch	340 MB	393× time / 172× vch
213	$(496, 9, 10, 406)$	6.74 h [33]	188.8 vch (28 vCPUs)	**138.8** s	**2.47** ch	358 MB	175× time / 76.5× vch
254	$(223, 8, 12, 127)$	16.0 h [34, 35]	8×GH200 (27.2 GPU-h*)	**30.4** s	**0.540** ch	313 MB	1,896× time (CPU vs. GPU)
252	$(1008, 10, 11, 898)$	375 s [33]	1.25 ch (12 cores)	**90.0** s	**0.025** ch	1.45 GB	4.17× time / 50.0× ch
147	$(509, 9, 11, 410)$	*No prior published solution*		**408** s	**7.25** ch	373 MB	**First-Blood** ($n' = 303$)
232	$(233, 8, 13, 129)$	*No prior published solution*		**332** s	**5.90** ch	254 MB	**First-Blood** ($n' = 233$)
245	$(493, 9, 12, 385)$	*No prior published solution*		**2510** s	**44.62** ch	422 MB	**First-Blood** ($n' = 326$)
172	$(1020, 10, 11, 910)$	*No prior published solution*		**89.9** s	**0.025** ch	1.51 GB	**First-Blood** (Route B)
165	$(251, 8, 14, 139)$	*No prior published solution*		**81.2** s	**1.44** ch	263 MB	**First-Blood** ($n' = 250$)
250	$(1010, 10, 12, 890)$	*No prior published solution*		**9.20** h†	**66,722** ch†	1.28 GB	**First-Blood** ($n' = 467$)
228	$(500, 9, 14, 374)$	*No prior published solution*		**57.1** min†	**2,463** ch†	468 MB	**First-Blood** ($n' = 361$)

Notes: Parameters are (n, m, t, k); ch, th, and vch denote physical core-hours (64 Xeon cores for Route A, 1 core for Route B), thread-hours [39], and vCPU-hours [33]. *16.0 h wall is from [35] and 27.2 GPU-h in [34] covers the two Krylov runs; †cluster runs (2,048–10,240 EPYC cores).

code rates and collapse once $R = k/n$ drops toward or below $1 - 2/t$. Even near this boundary, solving the resulting polynomial or MinRank systems required up to 9.47 h and 223 GB RAM on 72 Xeon cores (`TII-125` [2], CRYPTO 2026) or 47.75 h on 256 EPYC cores (`TII-223` [18]).

Primal Hold-Out Attacks. To reach moderate code rates where shortened-dual kernels vanish, Ghoshal, Ishai, Jain, and Sun [17] introduced the primal *hold-out distinguisher*, which Vedenev [38], Weis [39], and Saarinen [34] extended to key recovery. Viewing the columns of a shortened generator matrix (of dimension $k_\ell \in [35, 50]$) as points on a hidden rational curve, the attack holds out one or two anchor columns and searches for low-degree k_ℓ-variate polynomials (typically of degree 6) whose values and low-order Hasse derivatives vanish at all remaining columns. By univariate Hermite interpolation along the curve, any such polynomial must also vanish at the held-out columns, exposing the secret support inside the kernel. However, representing degree-6 multilinear polynomials in k_ℓ variables still produces sparse matrices of width $\binom{k_\ell}{6} \sim 10^7$ columns, turning kernel extraction into a large-scale sparse Krylov or block-Lanczos computation. Recovering `TII-253` $(214, 8, 9)$ took 21.7 h on 32 Xeon threads by Weis at Anthropic [39] or 14.0 h on an H200 GPU with 48 CPU threads [38] (which also solved the shorter instances `TII-173` and `TII-249`). Scaling further to `TII-254` $(223, 8, 12)$ required two deflated 15,527,170-column operators taking 16.0 h wall time and 128 GiB RAM on up to 8× NVIDIA GH200 GPUs (27.2 GH200 GPU-h across the two Krylov stages) [34, 35].

1.2 Main Results and Technical Overview

Our primary method (Route A) bypasses both the dual rate barrier and multi-million-column hold-out matrices by working directly with *minimum-weight codewords of a shortened primal code*. Although the RREF transformation $H_{\text{pub}} = T^{-1} H_{\text{raw}}(L, g)$ scrambles the rows of the parity-check matrix, the primal code $\Gamma(L, g) = \ker_{\mathbb{F}_2}(H_{\text{pub}})$ is unchanged. In characteristic 2, differentiating the support locator polynomial $\sigma_S(X) = \prod_{j \in S}(X - z_j)$ of any minimum odd-weight codeword ($w = 2t + 1$) annihilates odd elementary symmetric functions and yields the exact polynomial square $\sigma'_S(X) = \bar{g}(X)^2$ [31, 25, 16] (Theorem 1). Matching its X^{2t-2} coefficient

gives the constant quadratic identity

$$e_2(z_S) = \sum_{0 \le u < v < n} c_u c_v \, z_u z_v = \bar{g}_{t-1}^2 \in \mathbb{F}_q. \tag{2}$$

Expanding (2) into m coordinate bits over $\mathbb{F}_2$ turns every weight-$(2t+1)$ codeword c into m binary quadratic equations over the secret support products $z_u z_v$. Route A turns this identity into a fast key-recovery engine via two pairs of techniques:

- **Primal Shortening and Cache-Resident Multi-Target ISD (Section 3):** Restricting H_{pub} to a window of $n' \ll n$ columns preserves the Goppa polynomial $g(X)$ while compressing the number of quadratic interpolation unknowns $F(d) = \Theta((n' - mt)^2)$ by up to 6.59×. Because multi-target ISD harvests *any* $F(d)$ distinct weight-$(2t+1)$ codewords from the shortened pool, the $\binom{n'}{w}$ pool-size factor cancels the $\binom{n'}{w}$ denominator of the single-word ISD capture probability (Proposition 1). Furthermore, in the shortened multi-core regime, keeping single-stage Ball-Collision and L1d-bitmap-filtered Stern–Dumer tables inside per-core L1d/L2 caches avoids the DRAM bandwidth bottleneck of multi-layer BJMM.
- **Public Polarization Quotient and Systematic $\mathbb{F}_2$-Linear Lifting (Section 4):** After restricting to the odd-weight affine hyperplane and quotienting out the public combinatorial identity $\sum_{u<v} c_u c_v = \binom{2t+1}{2} \equiv t \pmod 2$ (Theorem 2), we transform the affine generator into systematic form $[I_d \mid P]$. This traps all d pivot diagonal weights on the main diagonal of the rank-m target matrices, allowing us to recover the shortened support L_A, extract $\bar{g}(X) = \sqrt{\sigma'_S(X)}$ from a single codeword, and extend L_A to all n columns via $O(n)$ closed-form field divisions without Gröbner bases.

For the single high-rate family $(m, t) = (10, 11)$, we pair Route A with a Frobenius-reduced shortened-dual HOVER solver (Route B, summarized in Section 4.4 and detailed in Appendix A).

As summarized in Table 1 and Section 5, our framework solves 94 official TII Track-2 challenges (`TII-83` through `TII-254`), achieving 55 first-blood recoveries (including `TII-133`, `TII-228` at $mt = 126$, and `TII-250` at $mt = 120$) and accelerating 39 published instances, with all 94 keys passing TII's official SageMath verification script (94/94 verified). The 92 single-node instances finish in 177.82 physical core-hours combined.

2 Binary Goppa Codes and Minimum-Weight Structure

2.1 Public-Key Construction over $\mathbb{F}_{2^m}$

Fix an extension degree $m \ge 1$ and field size $q = 2^m$, with $\mathbb{F}_q = \mathbb{F}_2[x]/(f(x))$ defined by a degree-m irreducible polynomial $f(x) \in \mathbb{F}_2[x]$ and primitive root $a = x \bmod f(x)$. Every element $\alpha = \sum_{b=0}^{m-1} c_b a^b \in \mathbb{F}_q$ has a dual representation:

- **Field scalar in $\mathbb{F}_q$:** α supports finite-field arithmetic and polynomial evaluation; in characteristic 2, $-1 = +1$, $(A+B)^2 = A^2 + B^2$, and the Frobenius squaring map $\alpha \mapsto \alpha^2$ is an automorphism with unique square-root inverse $\sqrt{\alpha} = \alpha^{2^{m-1}}$.
- **Bit vector in $\mathbb{F}_2^m$:** Via the $\mathbb{F}_2$-vector space isomorphism $\phi : \mathbb{F}_q \to \mathbb{F}_2^m$ mapping $\sum_{b=0}^{m-1} c_b a^b \mapsto (c_0, c_1, \ldots, c_{m-1})^\top$, α is a vertical column vector of m bits.

In the official TII specification [21], the public key H_{pub} is generated from the private key (L, g) in three steps:

Definition 1 (Binary Goppa Public-Key Construction)**.** Let $L = (z_0, z_1, \ldots, z_{n-1}) \in \mathbb{F}_q^n$ be an ordered tuple of $n \le q$ pairwise distinct support elements, and let $g(X) = \sum_{i=0}^{t} g_i X^i \in \mathbb{F}_q[X]$ be a square-free Goppa polynomial of degree $t \ge 2$ satisfying $g(z_j) \ne 0$ for all $0 \le j < n$ (chosen monic irreducible in TII's generator [21] and Classic McEliece).

1. **Step 1: Weighted Vandermonde Parity-Check Matrix over $\mathbb{F}_q$.** Form the $t \times n$ alternant parity-check matrix $H_{\mathbb{F}_q}(L, g) \in \mathbb{F}_q^{t \times n}$ with entries

$$H_{\mathbb{F}_q}(L, g)_{i,j} = \frac{z_j^i}{g(z_j)} \qquad (0 \le i < t,\ 0 \le j < n), \tag{3}$$

explicitly written as

$$H_{\mathbb{F}_q}(L, g) = \begin{pmatrix} \frac{1}{g(z_0)} & \frac{1}{g(z_1)} & \cdots & \frac{1}{g(z_{n-1})} \\ \frac{z_0}{g(z_0)} & \frac{z_1}{g(z_1)} & \cdots & \frac{z_{n-1}}{g(z_{n-1})} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{z_0^{t-1}}{g(z_0)} & \frac{z_1^{t-1}}{g(z_1)} & \cdots & \frac{z_{n-1}^{t-1}}{g(z_{n-1})} \end{pmatrix} \in \mathbb{F}_q^{t \times n}.$$

2. **Step 2: Vertical m-Bit Expansion into $\mathbb{F}_2^{mt \times n}$.** Expand each entry $H_{\mathbb{F}_q}(L, g)_{i,j} \in \mathbb{F}_q$ vertically into an m-bit column vector via $\phi : \mathbb{F}_q \to \mathbb{F}_2^m$, producing the raw binary parity-check matrix $H_{\text{raw}}(L, g) \in \mathbb{F}_2^{mt \times n}$:

$$H_{\text{raw}}(L, g)_{mi+b,j} = \phi\left(\frac{z_j^i}{g(z_j)}\right)_b \qquad (0 \le i < t,\ 0 \le b < m,\ 0 \le j < n). \tag{4}$$

The *binary Goppa code* $\Gamma(L, g) \subseteq \mathbb{F}_2^n$ is the binary right kernel of this matrix:

$$\Gamma(L, g) \coloneqq \left\{c \in \mathbb{F}_2^n : H_{\mathbb{F}_q}(L, g)\, c^\top = 0 \text{ in } \mathbb{F}_q^t\right\} = \ker_{\mathbb{F}_2}\big(H_{\text{raw}}(L, g)\big). \tag{5}$$

3. **Step 3: Row-Scrambling via RREF.** When $H_{\text{raw}}(L, g)$ has full row rank mt, the code has dimension $k = n - mt$. Key generation applies Gaussian elimination over $\mathbb{F}_2$ and publishes only the canonical reduced row-echelon form

$$H_{\text{pub}} = \text{RREF}\big(H_{\text{raw}}(L, g)\big) = T^{-1} H_{\text{raw}}(L, g) \in \mathbb{F}_2^{mt \times n}, \tag{6}$$

where $T \in \text{GL}_{mt}(\mathbb{F}_2)$ is the secret invertible $mt \times mt$ binary row-mixing matrix determined by the mt pivot columns of $H_{\text{raw}}(L, g)$ (for systematic instances, $H_{\text{pub}} = [I_{mt} \mid R]$).

2.2 Row Scrambling and Codeword Space Invariance

If an attacker were given the unscrambled matrix $H_{\text{raw}}(L, g)$ prior to Step 3, regrouping the first m rows of column j would yield $v_{0,j} = 1/g(z_j) \in \mathbb{F}_q^\times$ and regrouping rows m through $2m - 1$ would yield $v_{1,j} = z_j/g(z_j) \in \mathbb{F}_q$. Their ratio immediately reveals every support element in $O(n)$ field divisions:

$$z_j = \frac{v_{1,j}}{v_{0,j}} = \frac{z_j/g(z_j)}{1/g(z_j)} \qquad (0 \le j < n), \tag{7}$$

after which $g(X)$ follows by linear interpolation from $g(z_j) = 1/v_{0,j}$.

On the public key $H_{\text{pub}} = T^{-1} H_{\text{raw}}(L, g)$, however, the unknown matrix $T^{-1} \in \text{GL}_{mt}(\mathbb{F}_2)$ takes arbitrary $\mathbb{F}_2$-linear combinations across all mt binary rows, mixing the b-th bit of $1/g(z_j)$ with the b'-th bit of $z_j^{t-1}/g(z_j)$. Individual rows of H_{pub} no longer group into coherent m-bit field elements, and brute-forcing $T \in \text{GL}_{mt}(\mathbb{F}_2)$ costs $\approx 2^{(mt)^2}$. (Note that any column permutation P of $H_{\text{raw}}(L, g)$ merely reorders the support tuple $L \mapsto LP$, so T^{-1} is the sole algebraic barrier between H_{pub} and (L, g).)

Left-multiplying by an invertible matrix $T^{-1} \in \mathrm{GL}_{mt}(\mathbb{F}_2)$ leaves the right kernel completely invariant:

$$c \in \Gamma(L,g) \iff H_{\mathrm{raw}}(L,g)\,c^\top = 0 \iff T^{-1}H_{\mathrm{raw}}(L,g)\,c^\top = H_{\mathrm{pub}}\,c^\top = 0. \tag{8}$$

Every codeword $c \in \ker_{\mathbb{F}_2}(H_{\mathrm{pub}})$ is therefore unaffected by T^{-1} and obeys the exact algebraic identities of $H_{\mathbb{F}_q}(L,g)$. Moreover, for any affine automorphism $z \mapsto \alpha z + \beta$ ($\alpha \in \mathbb{F}_q^\times, \beta \in \mathbb{F}_q$) and Frobenius power $\sigma \in \mathrm{Gal}(\mathbb{F}_q/\mathbb{F}_2)$, transforming (L,g) to $(\alpha L^\sigma + \beta,\ \alpha^t g^\sigma(\alpha^{-1}(X-\beta)))$ preserves the binary parity-check row space and keeps g monic; any such equivalent key $(\hat{L},\hat{g})$ satisfies (1).

2.3 Minimum-Weight Locator Derivative Identity

To connect $\ker_{\mathbb{F}_2}(H_{\mathrm{pub}})$ to (L,g), observe that for each $j \in \{0,\dots,n-1\}$, the polynomial $g(X)-g(z_j)$ is divisible by $X-z_j$ with degree-$(t-1)$ quotient $\sum_{i=0}^{t-1} g_{i+1}\sum_{r=0}^{i} z_j^r X^{i-r}$. Hence, in $\mathbb{F}_q[X]/(g(X))$,

$$\frac{1}{X-z_j} \equiv -\frac{g(X)-g(z_j)}{g(z_j)(X-z_j)} \pmod{g(X)}.$$

Summing over $j = 0,\dots,n-1$ weighted by $c_j \in \{0,1\}$ shows that the t field parity checks $\sum_{j=0}^{n-1} c_j z_j^i / g(z_j) = 0$ ($0 \le i < t$) are equivalent to the classical *Goppa syndrome condition* [31, 25]:

$$S_c(X) \coloneqq \sum_{j=0}^{n-1} \frac{c_j}{X-z_j} = \sum_{j\in S} \frac{1}{X-z_j} \equiv 0 \pmod{g(X)}, \tag{9}$$

where $S = \mathrm{supp}(c) = \{j \in \{0,\dots,n-1\} : c_j = 1\}$ and $w = |S| = \mathrm{wt}(c)$. Define the *support locator polynomial* of c:

$$\sigma_S(X) \coloneqq \prod_{j\in S}(X-z_j) = \sum_{i=0}^{w} e_i(z_S)\,X^{w-i} \in \mathbb{F}_q[X], \tag{10}$$

where $e_i(z_S) = \sum_{J\subseteq S,|J|=i}\prod_{j\in J} z_j$ is the i-th elementary symmetric polynomial of the active support elements $z_S = (z_j)_{j\in S}$ (since $-1 = +1$ in $\mathbb{F}_q$). Let $\bar{g}(X) = g(X)/g_t = X^t + \sum_{i=0}^{t-1}\bar{g}_i X^i$ denote the monic normalization of $g(X)$. In binary square-free Goppa codes, divisibility $\bar{g}(X)^2 \mid \sigma_S'(X)$ yields $d_{\min} \ge 2t+1$ [31, 25], and Finiasz [16] observed that at minimum weight $w = 2t+1$ monic degree equality forces $\sigma_S'(X) = \bar{g}(X)^2$ (fixing all even elementary symmetric coefficients of $\sigma_S(X)$ when (L,g) is known; see [38, 39] for higher-order differential identities on general codewords). Route A exploits the X^{2t-2} coefficient of this classical identity in the inverse direction to build quadratic key-recovery equations over unknown (L,g):

Theorem 1 (Minimum-Weight Locator Derivative Identity)**.** *Let $\Gamma(L,g)$ be a binary Goppa code with square-free degree-t Goppa polynomial $g(X)$ and monic normalization $\bar{g}(X) = X^t + \sum_{i=0}^{t-1}\bar{g}_i X^i$. For every codeword $c \in \Gamma(L,g)$ of minimum odd weight $w = 2t+1$ with support $S = \mathrm{supp}(c)$, the formal derivative $\sigma_S'(X)$ satisfies the exact polynomial identity*

$$\sigma_S'(X) = \bar{g}(X)^2 = X^{2t} + \sum_{j=1}^{t} \bar{g}_{t-j}^2\,X^{2(t-j)}. \tag{11}$$

Consequently, for every $j \in \{1,\dots,t\}$, the even elementary symmetric function $e_{2j}(z_S)$ is independent of the codeword c and equals the constant field element

$$e_{2j}(z_S) = \sum_{\substack{J\subseteq S\\ |J|=2j}} \prod_{r\in J} z_r = \bar{g}_{t-j}^2. \tag{12}$$

In particular, for $j = 1$, every weight-$(2t+1)$ codeword $c \in \Gamma(L, g)$ satisfies the quadratic relation

$$e_2(z_S) = \sum_{0 \le u < v < n} c_u c_v \, z_u z_v = \bar{g}_{t-1}^2. \tag{13}$$

Proof. We proceed in four steps:

1. **Step 1 (Syndrome divisibility via $\sigma'_S(X)$):** Applying the product rule to $\sigma_S(X) = \prod_{j \in S}(X - z_j)$ yields

$$\sigma'_S(X) = \sum_{j \in S} \prod_{r \in S \setminus \{j\}} (X - z_r) = \sigma_S(X) \sum_{j \in S} \frac{1}{X - z_j} = \sigma_S(X)\, S_c(X).$$

   Because $g(z_j) \neq 0$ for all $j \in S$, $\gcd(\sigma_S(X), g(X)) = 1$. Thus $S_c(X) \equiv 0 \pmod{g(X)}$ holds if and only if $g(X) \mid \sigma'_S(X)$.

2. **Step 2 (Perfect-square structure of $\sigma'_S(X)$ in characteristic 2):** Over $\mathbb{F}_q$ ($\operatorname{char} \mathbb{F}_q = 2$), $(X^r)' = rX^{r-1}$ vanishes for even r and equals X^{r-1} for odd r. At odd weight $w = 2t + 1$, differentiating (10) annihilates all terms with even $w - i$ (odd i) and retains only even indices $i = 2j$ $(0 \le j \le t)$:

$$\sigma'_S(X) = \sum_{i=0}^{w} (w - i)\, e_i(z_S)\, X^{w-i-1} = \sum_{j=0}^{t} e_{2j}(z_S)\, X^{2(t-j)}. \tag{14}$$

   Since Frobenius squaring $a \mapsto a^2$ is a bijection on $\mathbb{F}_{2^m}$ with inverse $a \mapsto a^{2^{m-1}}$, the identity $(A + B)^2 = A^2 + B^2$ factors (14) as an exact square:

$$\sigma'_S(X) = h_S(X)^2, \qquad \text{where} \quad h_S(X) := \sum_{j=0}^{t} e_{2j}(z_S)^{2^{m-1}}\, X^{t-j} \in \mathbb{F}_q[X].$$

3. **Step 3 (Monic polynomial equality from square-freeness of $g(X)$):** By Step 1, $g(X) \mid h_S(X)^2$. Since $g(X)$ is square-free, its irreducible factors are pairwise distinct, so $g(X) \mid h_S(X)$ and hence $\bar{g}(X)^2 \mid h_S(X)^2 = \sigma'_S(X)$ (which implies $\deg(\sigma'_S) \ge 2t$ and $w \ge 2t + 1$). At $w = 2t + 1$, both $\sigma'_S(X)$ in (14) (with $e_0(z_S) = 1$) and $\bar{g}(X)^2 = X^{2t} + \sum_{j=1}^{t} \bar{g}_{t-j}^2 X^{2(t-j)}$ are monic of degree $2t$, so divisibility forces equality (11).

4. **Step 4 (Quadratic codeword identity):** Matching the coefficient of $X^{2(t-j)}$ between (14) and (11) yields $e_{2j}(z_S) = \bar{g}_{t-j}^2$. Setting $j = 1$ gives $e_2(z_S) = \sum_{u<v \in S} z_u z_v = \sum_{0 \le u < v < n} c_u c_v z_u z_v = \bar{g}_{t-1}^2$.

□

3 Primal Shortening and Multi-Target ISD

3.1 Primal Shortening and the ISD Cancellation Law

Full-Length Interpolation Bottleneck. In a full-length Goppa code $\Gamma(L, g)$ of dimension $k = n - mt$, writing $c = x^\top G$ in message coordinates $x \in \mathbb{F}_2^k$ (or $d_{\text{full}} = k - 1 = n - mt - 1$ affine coordinates on the odd-weight hyperplane) turns each of the m bit slices of (13) into a binary quadratic equation with

$$F(d_{\text{full}}) = 1 + d_{\text{full}} + \binom{d_{\text{full}}}{2} = \Theta\big((n - mt)^2\big) \tag{15}$$

quadratic monomials. Isolating the m secret relations on the full code requires harvesting at least $F(d_{\text{full}})$ distinct weight-$(2t + 1)$ codewords and performing dense Gaussian elimination over $\mathbb{F}_2$ in $O(F(d_{\text{full}})^3)$ time ($F(443) = 98{,}347$ on `TII-125`; $F(889) = 395{,}606$ on `TII-250`).

Quadratic Feature Compression via Primal Shortening. Instead of running ISD on all n columns of H_{pub}, we select a window $A \subset \{0, \ldots, n-1\}$ of $n' = |A| \ll n$ columns such that $H_A \coloneqq (H_{\text{pub}})_{:,A} \in \mathbb{F}_2^{mt \times n'}$ has full row rank mt, and clamp all $n - n'$ coordinates outside A to zero ($c_j = 0$ for $j \notin A$). In the Goppa syndrome (9), terms with $c_j = 0$ drop out:

$$\sum_{j=0}^{n-1} \frac{c_j}{X - z_j} = \sum_{j \in A} \frac{c_j}{X - z_j} \equiv 0 \pmod{g(X)}.$$

By standard Goppa shortening invariance [25, 22], the shortened primal code $\mathcal{C}_A = \ker_{\mathbb{F}_2}(H_A)$ is identically the binary Goppa code $\Gamma(L_A, g)$ of length n', dimension $k' = n' - mt$, restricted support $L_A = (z_j)_{j \in A}$, and the *unchanged* Goppa polynomial $g(X)$. Shortening reduces the odd-weight affine dimension from $d_{\text{full}} = n - mt - 1$ to $d = n' - mt - 1$: on `TII-125` $(507, 9, 7)$, choosing $n' = 237$ compresses $F(d)$ from 98,347 to **15,052** (6.53× fewer features and $\approx$ **278**× faster dense elimination); on `TII-250` $(1010, 10, 12)$, choosing $n' = 467$ compresses $F(d)$ from 395,606 to **60,032** (6.59× compression).

Single-Target vs. Multi-Target ISD. In syndrome decoding, a single target error vector e of weight w is fixed (or N target syndromes are given in Sendrier's multi-target DOOM setting [36]), so restricting to n' columns succeeds only if $\text{supp}(e) \subset A$, which occurs with tiny probability $\binom{n'}{w}/\binom{n}{w} \ll 1$. In Route A, by contrast, minimum-weight codeword search [24, 8] harvests *any* $F(d)$ distinct weight-$(2t+1)$ codewords in $\mathcal{C}_A$. Under the random-code heuristic, the expected supply of weight-w codewords in $\mathcal{C}_A$ and the single-word ISD capture probability exhibit an exact combinatorial cancellation:

Proposition 1 (Feature Count, Heuristic Word Supply, and Multi-Target ISD Cancellation)**.** *Let $H_A \in \mathbb{F}_2^{mt \times n'}$ be a full-rank shortened parity-check matrix with shortened dimension $k' = n' - mt$, odd-weight affine dimension $d = k' - 1 = n' - mt - 1$, and target weight $w = 2t + 1$.*

1. ***Required Evaluation Rank and Heuristic Word Supply:*** *The odd-weight affine chart has $F(d) = 1 + d + \binom{d}{2}$ Boolean quadratic features. Isolating a relation kernel of dimension q_{ker} requires at least $F(d) - q_{\text{ker}}$ independent evaluation rows; using the observed kernel profile $q_{\text{obs}}(m, t)$ (Proposition 2) as a planning target gives*

$$W_{req}(n', m, t) = F(d) - q_{\text{obs}}(m, t). \tag{16}$$

Under the uniform random parity-check heuristic, linearity of expectation gives the expected weight-w codeword supply

$$W_{avail}(n', m, t) \approx \binom{n'}{w} 2^{-mt}. \tag{17}$$

For a safety factor $\gamma \geq 1$ (e.g., $\gamma = 1.4$), define the planning window and minimal window $n'_{\text{min}}(m, t) \coloneqq n'_{\text{plan}}(m, t; 1)$ by

$$n'_{\text{plan}}(m, t; \gamma) = \min\left\{ n' > mt + 2m : \binom{n'}{2t+1} 2^{-mt} \geq \gamma \, W_{req}(n', m, t) \right\}. \tag{18}$$

2. ***Multi-Target ISD Cancellation Law:*** *In one RREF iteration of a degree-p ISD algorithm on H_A, the n' columns are partitioned into mt pivot columns and $k' = n' - mt$ non-pivot columns. A fixed weight-w codeword of $\mathcal{C}_A$ has weight p on the k' non-pivot columns and weight $w - p$ on the mt pivot columns with hypergeometric probability*

$$P_1(n') = \frac{\binom{k'}{p}\binom{mt}{w-p}}{\binom{n'}{w}}. \tag{19}$$

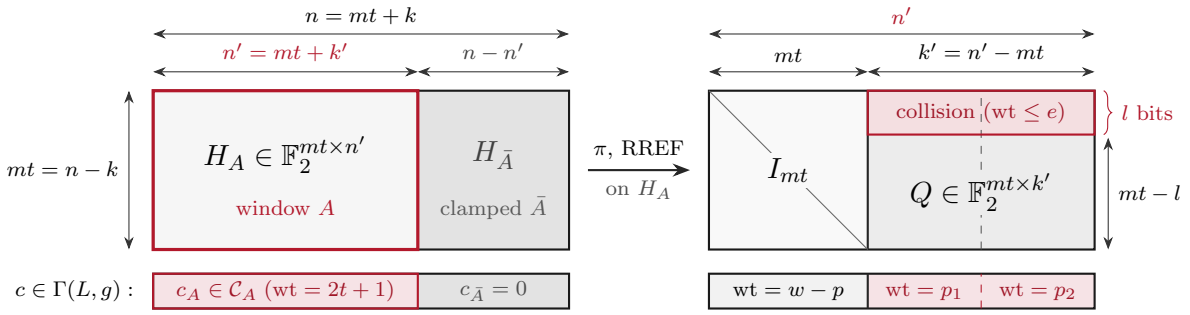

Figure 1: Matrix structure of primal shortening and multi-target ISD for a Goppa code of length n, codimension mt, and dimension $k = n - mt$. Left: permuting a full-rank n'-column window A to the left and clamping the remaining $n - n'$ columns of H_{pub} to zero restricts the search to $H_A \in \mathbb{F}_2^{mt \times n'}$ ($k' = n' - mt$), whose right kernel $\mathcal{C}_A = \Gamma(L_A, g)$ preserves $g(X)$. Right: each ISD round reduces $H_A\pi$ to $[I_{mt} \mid Q]$ and collides non-pivot half-lists of weights $p_1 + p_2 = p$ on an l-bit window to harvest codewords of odd weight $w = 2t + 1$.

By linearity of expectation across all $W_{avail}(n', m, t)$ weight-w codewords in $\mathcal{C}_A$, the expected number of weight-w codewords captured per ISD *iteration in the unsaturated regime is*

$$\text{Yield}(n') = W_{avail}(n', m, t) \cdot P_1(n') \approx \frac{\cancel{\binom{n'}{w}}}{2^{mt}} \cdot \frac{\binom{n'-mt}{p}\binom{mt}{w-p}}{\cancel{\binom{n'}{w}}} = \frac{\binom{n'-mt}{p}\binom{mt}{w-p}}{2^{mt}}, \tag{20}$$

where the $\binom{n'}{w}$ terms cancel identically.

Proof. For Part 1, rank-nullity requires evaluation rank $F(d) - q_{\text{ker}}$ for a kernel of dimension q_{ker}, and in a uniform random $mt \times n'$ binary matrix each of the $\binom{n'}{w}$ weight-w vectors satisfies the mt checks with probability 2^{-mt}. For Part 2, a fixed weight-w support meets a random (k', mt) partition in $(p, w-p)$ columns with hypergeometric probability $P_1(n')$; summing over the $W_{\text{avail}}(n', m, t)$ expected codewords gives (20). Transfer of the random-code supply expectation to Goppa codes and the choice of γ are planning heuristics rather than universal theorems. □

End-to-End Speedup near n'_{plan}. Equation (20) shows that the expected yield per inner-loop syndrome check is $\text{Yield}(n')/\binom{k'}{p} = \binom{mt}{w-p}2^{-mt}$, which is independent of n'. Meanwhile, reducing n' toward n'_{plan} simultaneously reduces the required codeword count $W_{\text{req}}(n', m, t) = \Theta(k'^2)$, the per-iteration RREF cost $O((mt)^2 n'/64)$, and the dense linear algebra cost $O(F(d)^3) = O(k'^6)$, as illustrated in Figure 1. Table 2 lists our chosen shortening windows across representative TII parameter families.

3.2 Cache-Resident Collision ISD vs. Multi-Layer BJMM

Single-Stage Collision ISD vs. Multi-Layer BJMM. For asymptotic parameters ($n = 3488, w = 129$), multi-layer representation decoders such as MMT [26] and BJMM [3] (and nearest-neighbor or sieving variants [27, 13]) reduce the asymptotic exponent by splitting a weight-p vector into overlapping halves across a multi-level merge tree. In the shortened TII regime ($mt \in [35, 126]$, $w \in [11, 29]$, and $k' = n' - mt \in [60, 347]$), however, single-stage collision ISD outperforms multi-layer BJMM for two concrete reasons:

1. **Compact Half-List Combinatorics:** After primal shortening, each half-list has only $k'/2 \in [30, 173]$ columns, so even full triple enumeration ($p_1 = p_2 = 3$) has at most $\binom{173}{3} = 847{,}549$ entries (and $\binom{118}{3} = 266{,}916$ on `TII-228`). At $k'/2 \approx 115$, adding $\varepsilon = 1$

Table 2: Shortening windows, codeword-supply estimates, and quadratic feature compression across representative TII parameter families.

Family	Weight	Full n	Short n'	Dim d	Required	Available	Compression
$(7,7)$	15	112–123	112	62	1,946	2.79×10^3	1.38×
$(8,5)$	11	252–253	143	102	5,237	7.85×10^3	**4.27**× (5.3k vs. 22.4k)
$(8,6)$	13	253–254	152	103	5,348	7.78×10^3	**3.93**× (5.4k vs. 21.1k)
$(8,7)$	15	221–254	163	106	5,663	8.32×10^3	**3.42**×
$(8,8)$	17	217–256	175	110	6,097	9.25×10^3	**2.99**×
$(8,9)$	19	214–255	187	114	6,547	9.88×10^3	**2.54**× (`TII-253`, 1.80 s)
$(8,10)$	21	226–255	199	118	7,013	1.02×10^4	**2.16**×
$(8,12)$	25	223–244	223	126	7,993	1.03×10^4	Solves `TII-254` in 30.4 s
$(8,14)$	29	246–256	246	133	8,903	8.49×10^3	Illustrative window
$(9,7)$	15	482–508	237	173	15,042	2.20×10^4	**6.56**× (15k vs. 98k)
$(9,8)$	17	495–512	253	180	16,281	2.45×10^4	**5.92**×
$(9,9)$	19	492–509	269	187	17,569	2.60×10^4	**5.19**×
$(9,10)$	21	496–509	286	195	19,101	2.86×10^4	**4.52**×
$(9,12)$	25	493–505	326	217	23,644	5.23×10^4	**3.31**×
$(9,14)$	29	500	361	234	27,486	6.16×10^4	**2.53**× (`TII-228`)
$(10,12)$	25	1010	467	346	60,021	1.36×10^5	**6.59**× (`TII-250`, 60k vs. 396k)
$(12,64)$	129	3488 (standard)	3488	2719	Unverified	$\approx 2^{23.47}$	Not attacked here

Notes: For each family (m,t) of target weight $w = 2t+1$, n' is the shortened length, $d = n' - mt - 1$ is the affine dimension, $W_{\text{req}} = F(d) - q_{\text{obs}}(m,t)$ and $W_{\text{avail}} \approx \binom{n'}{w} 2^{-mt}$ are the required and heuristic available codewords, and 'Compression' is $F(n_{\max} - mt - 1)/F(d)$ across each family's 'Full n' range.

Table 3: Multi-target ISD modes, window collision parameters, and per-thread cache footprints across TII parameter regimes.

Mode	Regime	Parameters	Left Step	Right Step	Cache Footprint
Mode 3	$mt < 88$ or $w < 21$	$p = 3$ (no split)	Direct triple loop	Word-0 popcount prune	$<$ **8** KB (L1d-resident)
Mode 5	$mt \in [88, 111], w \geq 21$	$p = 2{+}3$, $l{=}15$, $e{=}1$	16× ball expand	1× bucket lookup	**128** KB bucket heads (L2)
Mode 6	$mt \geq 112$, high k' (`250`)	$p = 3{+}3$, $l{=}17$, $e{=}0$	1× insert + bitmap bit	1-bit L1d bitmap gate	**16** KB bitmap in **L1d**
Mode 7	$mt \geq 112$, mod. k' (`228`)	$p = 3{+}3$, $l{=}15$, $e{=}1$	1× exact insert	16× ball probe	**128** KB heads + 5.7 MB pool

Notes: 'Parameters' list the non-pivot weight split $p = p_1 + p_2$, collision window size l (in bits), and allowed window Hamming weight e (ball radius).

cancellation bits in a 3-level BJMM tree provides minimal representation gain while introducing intermediate hash-join stages and duplicate filters that cost 5–10× more instructions per candidate than a register-level 3-XOR loop.

2. **Per-Core L1d/L2 Cache Residency vs. Shared DRAM Bandwidth:** Across 64 physical Xeon cores (or 2,048–10,240 physical EPYC cores on cluster runs), all cores share the socket's DRAM memory controllers. Multi-megabyte per-thread BJMM hash tables spill out of L2 (1.25 MB/core) and L3 cache into DRAM, stalling cores on memory bus contention. Keeping each thread's hot collision structure inside private L1d (48 KB) or L2 (1.25 MB) cache allows all cores to run at full clock speed with linear multi-core scaling.

Cache-Resident Multi-Target ISD Design. In each round, a worker thread permutes the columns of $H_A \in \mathbb{F}_2^{mt \times n'}$ (with incremental pivot updates [8]), computes a 64-bit word-packed RREF to obtain $\tilde{H}_A = [I_{mt} \mid Q]$ ($Q \in \mathbb{F}_2^{mt \times k'}$), and searches for weight-p non-pivot vectors $v \in \mathbb{F}_2^{k'}$ whose syndrome $u = Qv \in \mathbb{F}_2^{mt}$ has weight $\text{wt}(u) = w - p$. Our sampler incorporates five architectural optimizations (Table 3):

1. **Bit-Transposed Non-Pivot Syndromes:** Immediately after RREF, each thread transposes the k' columns of Q into vertical 64-bit words ($\lceil mt/64 \rceil \leq 2$ words per column for all

$mt \leq 126$), reducing column additions to 1–2 register-level XORs and weight checks to 1–2 hardware population-count instructions.

2. **Mode 3: Word-0 Early-Pruned Lee–Brickell ($p = 3$):** For $mt < 88$ or $w < 21$ (68 of 89 Route A instances), enumerating non-pivot triples ($a < b < c$) without hash tables is fastest. When $mt > 64$, the inner loop checks whether the Hamming weight of the first 64-bit syndrome word s_0 already exceeds $w - 3$ before computing the second word s_1, pruning $> 99\%$ of triples in 2 clock cycles.

3. **Modes 5 & 7: Asymmetric Read/Write Ball-Collision ($l = 15, e = 1$):** For $mt \geq 88, w \geq 21$, we partition the k' non-pivot columns into blocks of sizes n_1 and $n_2 = k' - n_1$ and allow ball collisions of weight $\mathrm{wt}(h_L \oplus h_R) \leq e = 1$ [6] on a 15-bit window ($2^{15} = 32{,}768$ buckets, whose 128 KB bucket-head array fits inside L2 cache). Because a 15-bit ball of radius 1 contains $1 + 15 = 16$ patterns, each RREF round captures $16\times$ more codewords than strict Stern–Dumer ($e = 0$). We flip the $16\times$ ball expansion between left insertion and right probing according to list size:

   - **Mode 5 ($p = 2 + 3 = 5$, e.g., `TII-254`):** Setting $n_1 = \min(145, \lfloor 0.42k' \rfloor)$, the asymmetric split makes the left pair list ($\binom{53}{2} = 1{,}378$ on `TII-254`) $47\times$ smaller than the right triple list ($\binom{74}{3} = 64{,}824$). Mode 5 expands $16\times$ on left insertion ($\leq 1.67 \times 10^5$ nodes across all families), so the hot right loop performs a single lookup per triple.
   - **Mode 7 ($p = 3 + 3 = 6$, e.g., `TII-228`):** With balanced halves $n_1 = \lfloor k'/2 \rfloor$, both sides enumerate triples ($\binom{118}{3} \approx 2.67 \times 10^5$), where $16\times$ left expansion would spill to DRAM. Mode 7 inserts once on the left (5.7 MB pool) and probes 16 buckets on the right, keeping the 128 KB bucket-head array in L2 cache.

4. **Mode 6: 16 KB L1d-Bitmap Pre-Filtered Stern–Dumer ($p = 3 + 3 = 6, l = 17$):** On `TII-250` ($k' = 347$, $\binom{173}{3} = 847{,}549$ triples/side), Mode 6 uses a 17-bit window (131,072 buckets, ≈ 8 MB per thread) guarded by a 16 KB occupancy bitmap (2,048 64-bit words) that stays 100% resident in the 48 KB L1d cache. Left insertions set bit h in the bitmap and record h in a compact dirty-bucket list; right-side triples test bit h in L1d first to reject empty buckets without touching the 8 MB bucket table, and only dirty buckets are reset at the end of the round.

5. **Lock-Free Atomic Global Deduplication:** All threads insert discovered weight-w codewords into a shared 2^{22}-slot open-addressing hash table via 64-bit FNV-1a support hashes and atomic compare-and-swap (CAS) operations, terminating as soon as the shared counter reaches W_{target}.

4 Quotient Reduction and Systematic Linear Lifting

4.1 Affine Chart and Public Polarization Quotient

Given $W \geq W_{\text{req}}(n', m, t)$ harvested codewords $c \in \mathcal{C}_A$ of odd weight $w = 2t + 1$, three structural properties govern the quadratic evaluation system (Theorem 2):

1. **Odd-Weight Affine Hyperplane Parameterization ($d = k' - 1$):** Every harvested codeword has odd weight $w = 2t + 1$ and satisfies the linear parity constraint $\sum_{u \in A} c_u \equiv 1 \pmod 2$. In message coordinates $c = x^\top G_A$ ($x \in \mathbb{F}_2^{k'}$), this reads $\ell \cdot x = 1$ where $\ell = G_A \mathbf{1}_{n'}$. Interpolating quadratics directly in $x \in \mathbb{F}_2^{k'}$ would introduce k' trivial parity-multiplied syzygies $x_i(\ell \cdot x + 1) = 0$. Parameterizing the affine hyperplane $\ell \cdot x = 1$ by $d = k' - 1$ free coordinates $y \in \mathbb{F}_2^d$ via

$$c(y) = c_0 + y^\top G_0 \qquad (y \in \mathbb{F}_2^d,\ d = k' - 1 = n' - mt - 1), \tag{21}$$

where $\mathrm{wt}(c_0)$ is odd and every row of $G_0 \in \mathbb{F}_2^{d\times n'}$ has even weight, removes all k' parity syzygies up front.

2. **Rank-$\leq m$ Multiplication Forms Up to Diagonal Correction:** Unlike dual-code quadratic-form attacks over $\mathbb{F}_{2^m}$ that target rank-2 alternating matrices [11, 2, 7], Route A pulls (13) back over $\mathbb{F}_2$ via linear functionals $F : \mathbb{F}_q \to \mathbb{F}_2$, giving $\sum_{u<v} c_u c_v F(z_u z_v) = F(\bar{g}_{t-1}^2)$. Because the sum ranges over strictly ordered pairs $u < v$, its symmetric coefficient matrix $M_F \in \mathbb{F}_2^{n'\times n'}$ has off-diagonal entries $(M_F)_{uv} = F(z_u z_v)$ $(u \neq v)$ and zeros on the diagonal $((M_F)_{uu} = 0)$. By contrast, the full multiplication matrix $\widetilde{M}_F \in \mathbb{F}_2^{n'\times n'}$ with $(\widetilde{M}_F)_{uv} = F(z_u z_v)$ for all u, v (including the diagonal $(\widetilde{M}_F)_{uu} = F(z_u^2) =: (w_F)_u$) factors through the m-dimensional space $\mathbb{F}_q$ and has $\mathrm{rank}_{\mathbb{F}_2}(\widetilde{M}_F) \leq m$. Since $\widetilde{M}_F = M_F + \mathrm{diag}(w_F)$, pulling back by G_0 yields an off-diagonal bilinear polarization matrix $B_F = G_0 M_F G_0^\top \in \mathbb{F}_2^{d\times d}$ such that adding $G_0\,\mathrm{diag}(w_F)G_0^\top$ drops its rank to $\leq m$:

$$L_F \coloneqq B_F + G_0\,\mathrm{diag}(w_F)G_0^\top = G_0 \widetilde{M}_F G_0^\top, \qquad \mathrm{rank}_{\mathbb{F}_2}(L_F) \leq m.$$

3. **Public Combinatorial Polarization $T_{\mathbf{pub}}$:** Every binary vector $c \in \{0,1\}^{n'}$ of constant weight $w = 2t+1$ satisfies the combinatorial identity

$$\sum_{0\leq u<v<n'} c_u c_v = \binom{2t+1}{2} \equiv t \pmod 2, \tag{22}$$

independent of (L, g). Substituting $c(y) = c_0 + y^\top G_0$ into (22) yields a public quadratic relation with bilinear polarization matrix

$$T_{\mathrm{pub}} = G_0(J_{n'} + I_{n'})G_0^\top \in \mathbb{F}_2^{d\times d}, \tag{23}$$

where $J_{n'} + I_{n'}$ has 1 off the diagonal and 0 on the diagonal. Projecting out T_{pub} leaves an m-dimensional quotient space spanned by the m secret bit-slice forms $B_1, \ldots, B_m$.

Theorem 2 (Primal Shortening Invariance, Quotient Reduction, and Support Extension)**.** *Let $\Gamma(L, g) = \ker_{\mathbb{F}_2}(H_{pub})$ be a binary Goppa code of length n and dimension $k = n - mt$, and let $A \subset \{0, \ldots, n-1\}$ be a subset of n' coordinate indices such that $H_A \coloneqq (H_{pub})_{:,A} \in \mathbb{F}_2^{mt\times n'}$ has full row rank mt.*

1. ***Shortened Goppa Invariance [25, 22]:*** *The shortened primal code $\mathcal{C}_A \coloneqq \{c_A \in \mathbb{F}_2^{n'} : H_A c_A^\top = 0\}$ is identically the binary Goppa code $\Gamma(L_A, g)$ of length n', dimension $k' = n' - mt$, and minimum distance $\geq 2t+1$, where $L_A = (z_j)_{j\in A}$ and $g(X)$ is the unchanged Goppa polynomial.*

2. ***Odd-Weight Affine Hyperplane and Bit-Sliced Polarization:*** *Let $G_A \in \mathbb{F}_2^{k'\times n'}$ be a generator matrix of $\mathcal{C}_A$ and $\ell = G_A \mathbf{1}_{n'} \in \mathbb{F}_2^{k'}$. Parameterizing the odd-weight hyperplane $\ell \cdot x = 1$ by $x(y) = e_{i_0} + Ey$ for $y \in \mathbb{F}_2^d$ $(d = k' - 1$, where $\ell_{i_0} = 1$ and $\mathrm{im}(E) = \ker(\ell^\top))$ and setting $G_0 = E^\top G_A \in \mathbb{F}_2^{d\times n'}$ and $c_0 = (G_A)_{i_0,:}$, every $\mathbb{F}_2$-linear functional $F \in \mathbb{F}_q^\vee$ pulls $e_2(z_S) = \bar{g}_{t-1}^2$ back to an affine quadratic relation $q_F(y) = 0$ on $y \in \mathbb{F}_2^d$ whose symmetric bilinear polarization matrix $B_F \in \mathbb{F}_2^{d\times d}$ $(B_{F,ii} = 0)$ satisfies*

$$B_F = G_0\, M_F\, G_0^\top, \qquad L_F \coloneqq B_F + G_0\,\mathrm{diag}(w_F)G_0^\top = G_0\, \widetilde{M}_F\, G_0^\top, \qquad \mathrm{rank}_{\mathbb{F}_2}(L_F) \leq m, \tag{24}$$

where $(M_F)_{uv} = F(z_u z_v)$ for $u \neq v$ with $(M_F)_{uu} = 0$, $(w_F)_u = F(z_u^2)$, and $(\widetilde{M}_F)_{uv} = F(z_u z_v)$ for all $u, v \in A$ has $\mathrm{rank}_{\mathbb{F}_2}(\widetilde{M}_F) \leq m$.

3. ***Public Exceptional Direction Elimination:*** *On every weight-$(2t+1)$ codeword $c \in \mathcal{C}_A$, the identity $\sum_{u<v} c_u c_v = \binom{2t+1}{2} \equiv t \pmod 2$ induces a public quadratic relation on $y \in \mathbb{F}_2^d$ with polarization matrix $T_{pub} = G_0(J_{n'}+I_{n'})G_0^\top \in \mathbb{F}_2^{d\times d}$. When $T_{pub} \neq 0$, it can be quotiented out; if the m secret relations are independent of each other and of T_{pub}, their classes span an m-dimensional subspace of the quotient.*

4. ***Closed-Form $O(n)$ Support Extension Outside*** *A [22]: Assume $t \geq 2$ and that the recovered $(L_A, \bar{g})$ is compatible with a representation of the full public code. Choosing mt independent columns $P_0 \subset A$, the unique basis change $T = H_{raw}(L_A, \bar{g})_{:,P_0}(H_A)^{-1}_{:,P_0} \in \mathrm{GL}_{mt}(\mathbb{F}_2)$ unscrambles the entire public key $\widetilde{H} = TH_{pub}$, and reading the first two m-bit field slices $v_{0,j}, v_{1,j} \in \mathbb{F}_q$ of each column $j \notin A$ yields [22, 38, 39, 33, 34]*

$$z_j = \frac{v_{1,j}}{v_{0,j}} \qquad (\forall\, j \in \{0,\dots,n-1\} \setminus A). \tag{25}$$

Proof. Part 1: A vector $c \in \mathbb{F}_2^n$ with $c_j = 0$ for $j \notin A$ satisfies $H_{\mathrm{pub}}c^\top = 0 \iff H_A c_A^\top = 0$, and its Goppa syndrome $\sum_{j\in A} c_j/(X - z_j) \equiv 0 \pmod{g(X)}$ is the defining condition of $\Gamma(L_A, g)$. Since $\mathrm{rank}_{\mathbb{F}_2}(H_A) = mt$ and $g(X)$ is square-free of degree t, $\dim_{\mathbb{F}_2} \mathcal{C}_A = n' - mt = k'$ and $d_{\min}(\mathcal{C}_A) \geq 2t+1$.

Part 2: Expanding $c_u(y)c_v(y)$ with $c_u(y) = c_{0,u} + \sum_{a=1}^d y_a (G_0)_{au}$ inside the quadratic relation $q_F(y) = \sum_{u<v} c_u(y)c_v(y)F(z_u z_v) + F(\bar{g}_{t-1}^2) = 0$ over $\mathbb{F}_2$, the diagonal quadratic terms $y_a^2 = y_a$ merge into the linear part of $q_F(y)$, while for $a < b$ the coefficient of $y_a y_b$ is

$$(B_F)_{ab} = \sum_{u<v} \big((G_0)_{au}(G_0)_{bv} + (G_0)_{bu}(G_0)_{av}\big)(M_F)_{uv} = \sum_{u\neq v} (G_0)_{au}(M_F)_{uv}(G_0)_{bv} = (G_0 M_F G_0^\top)_{ab}.$$

Because M_F is symmetric with zero diagonal, $(G_0 M_F G_0^\top)_{aa} = \sum_{u<v}(G_0)_{au}(G_0)_{av}((M_F)_{uv} + (M_F)_{vu}) = 0$ over $\mathbb{F}_2$, so $B_F = G_0 M_F G_0^\top$ holds on all entries. Next, fix an $\mathbb{F}_2$-basis $e_1, \dots, e_m$ of $\mathbb{F}_q$ with dual coordinates $e_1^*, \dots, e_m^* \in \mathbb{F}_q^\vee$ so $z_u = \sum_{r=1}^m e_r^*(z_u) e_r$. By $\mathbb{F}_2$-linearity of F, $(\widetilde{M_F})_{uv} = F(z_u z_v) = \sum_{r=1}^m e_r^*(z_u) F(e_r z_v)$ is a sum of m rank-1 outer products over $\mathbb{F}_2$, so $\mathrm{rank}_{\mathbb{F}_2}(\widetilde{M_F}) \leq m$ and $L_F = G_0(M_F + \mathrm{diag}(w_F))G_0^\top = G_0 \widetilde{M_F} G_0^\top$ has rank $\leq m$.

Part 3: Every weight-$(2t+1)$ vector has $\binom{2t+1}{2} = t(2t+1) \equiv t \pmod 2$ active pairs $u < v$. Replacing M_F in Part 2 by the off-diagonal all-ones matrix $J_{n'} + I_{n'}$ yields the polarization $T_{\mathrm{pub}} = G_0(J_{n'} + I_{n'})G_0^\top$.

Part 4: Under compatibility, $TH_{\mathrm{pub}} = H_{\mathrm{raw}}(L, \bar{g})$ is uniquely determined by inverting its restriction to the mt independent columns $P_0 \subset A$. Its first two m-bit field slices in column $j \notin A$ equal $v_{0,j} = 1/\bar{g}(z_j)$ and $v_{1,j} = z_j/\bar{g}(z_j)$, whose ratio is z_j. Full row-space equality $\mathrm{RREF}(H_{\mathrm{raw}}(\hat{L}, \bar{g})) = H_{\mathrm{pub}}$ is verified before accepting the extended key. □

Proposition 2 (Conditional Relation-Space Identification)**.** *Let E evaluate the $F(d) = 1 + d + \binom{d}{2}$ Boolean affine quadratic monomials on the harvested weight-$(2t+1)$ words, and let U be the span of the m secret quadratic relations and the public relation. If these $m+1$ relations are independent and $\mathrm{rank}_{\mathbb{F}_2}(E) = F(d) - (m+1)$, then $\ker E = U$ (cf. [12] on concrete rank verification in structured algebraic code systems). Across our solved Route A instances, the observed kernel dimension follows the profile*

$$q_{\mathrm{obs}}(m,t) = \begin{cases} 2m+1 & \text{if } t = 5, \\ m+1 & \text{if } t \geq 4 \text{ and } t \neq 5. \end{cases} \tag{26}$$

For the 84 solved Route A instances with $t \neq 5$, quotienting $\ker E$ by T_{pub} leaves an m-dimensional quotient space. For the 5 solved instances with $t = 5$ (where analogous $t = 5$ degeneracies and extra kernel relations were independently observed on the shortened-dual and hold-out sides in [2, 39]), quotienting by T_{pub} leaves a $2m$-dimensional quotient; a Gray-code rank census and subspace probe isolate the true m-dimensional subspace in < 0.2 s.

4.2 Systematic Linear Lifting and Support Extension

Given m basis matrices $B_1, \ldots, B_m \in \mathbb{F}_2^{d\times d}$ representing the m-dimensional secret quotient space (since each row of G_0 has even weight, $G_0\mathbf{1}_{n'} = 0$ and $(G_0G_0^\top)_{aa} = 0$, so $T_{\text{pub}} = G_0\,\mathrm{diag}(\mathbf{1}_{n'})G_0^\top$ and representatives differing by T_{pub} simply shift $w^{(r)} \mapsto w^{(r)} + \mathbf{1}_{n'}$), each $r \in \{1, \ldots, m\}$ admits a binary diagonal weight vector $w^{(r)} \in \mathbb{F}_2^{n'}$ ($(w^{(r)})_u = F_r(z_u^2)$) such that

$$L^{(r)} = B_r + G_0\,\mathrm{diag}(w^{(r)})G_0^\top \quad \text{has} \quad \mathrm{rank}_{\mathbb{F}_2}(L^{(r)}) = m. \tag{27}$$

Decoupling Pivot Weights via Systematic Coordinates $[I_d \mid P]$. Whereas generic MinRank modelings such as Kipnis–Shamir [20] and Support-Minors [1] produce bilinear systems solved via Gröbner bases or XL, the unknowns $w^{(r)}$ here enter only through a diagonal matrix sandwiched by the known public matrix G_0. Since $G_0 \in \mathbb{F}_2^{d\times n'}$ has full row rank d, an invertible row transformation $U \in \mathrm{GL}_d(\mathbb{F}_2)$ and column permutation put G_0 into systematic form:

$$G_s \coloneqq UG_0 = \left[\, I_d \;\middle|\; P \,\right] \in \mathbb{F}_2^{d\times n'}, \tag{28}$$

where I_d is the $d \times d$ identity on the d pivot columns and $P \in \mathbb{F}_2^{d\times(mt+1)}$ is the non-pivot block on the remaining $n' - d = mt + 1$ columns. We split $w^{(r)} = (u^{(r)}, z^{(r)})$ into d pivot weights $u^{(r)} \in \mathbb{F}_2^d$ and $mt + 1$ non-pivot weights $z^{(r)} \in \mathbb{F}_2^{mt+1}$. In the transformed basis $B_r \leftarrow UB_rU^\top$, the diagonal pullback decomposes as

$$\left[\, I_d \;\middle|\; P \,\right] \begin{pmatrix} \mathrm{diag}(u^{(r)}) & 0 \\ 0 & \mathrm{diag}(z^{(r)}) \end{pmatrix} \begin{pmatrix} I_d \\ P^\top \end{pmatrix} = \mathrm{diag}(u^{(r)}) + P\,\mathrm{diag}(z^{(r)})P^\top.$$

Hence the rank-m target matrix $L^{(r)} \in \mathbb{F}_2^{d\times d}$ takes the exact structural form

$$L^{(r)} = \underbrace{B_r + P\,\mathrm{diag}(z^{(r)})P^\top}_{\text{controls all off-diagonal entries } (i\neq j)} + \underbrace{\mathrm{diag}(u^{(r)})}_{\text{only on the main diagonal } (i=j)}, \qquad \mathrm{rank}_{\mathbb{F}_2}(L^{(r)}) = m. \tag{29}$$

Because all d pivot unknowns $u^{(r)}$ are trapped on the main diagonal (Figure 2), every off-diagonal entry ($i \neq j$) of $L^{(r)}$ depends *only* on the $mt+1$ non-pivot weights $z^{(r)}$. This enables a three-step purely $\mathbb{F}_2$-linear lift:

1. **Step 1: Off-Diagonal Kernel Linearization for $z^{(1)}$.** Partition $\{1, \ldots, d\}$ into a row block R of size $k_R > m$ (e.g., $k_R = 30$) and its complement $J = \{1, \ldots, d\} \setminus R$ ($|J| = d - k_R$). Because $R \cap J = \emptyset$, the $k_R \times |J|$ submatrix $L^{(1)}_{R,J}$ lies entirely off the main diagonal, where $\mathrm{diag}(u^{(1)})$ vanishes:

$$L^{(1)}_{R,J} = (B_1)_{R,J} + P_{R,:}\,\mathrm{diag}(z^{(1)})P_{J,:}^\top \in \mathbb{F}_2^{k_R\times|J|}. \tag{30}$$

When $\mathrm{rank}_{\mathbb{F}_2}(L^{(1)}_{R,J}) = m$, its left kernel has dimension $k_R - m$: any left-kernel vector $\alpha \in \mathbb{F}_2^{k_R}$ satisfies $\alpha^\top(B_1)_{R,J} + (\alpha^\top P_{R,:})\,\mathrm{diag}(z^{(1)})P_{J,:}^\top = 0$. Absorbing $z^{(1)}$ into the auxiliary vector $y = (\alpha^\top P_{R,:}) \odot z^{(1)} \in \mathbb{F}_2^{mt+1}$ ($y_h = (\alpha^\top P_{R,:})_h \cdot z_h^{(1)}$) produces the homogeneous linear system

$$\alpha^\top(B_1)_{R,J} + y^\top P_{J,:}^\top = 0 \iff \left[\, (B_1)_{R,J}^\top \;\middle|\; P_{J,:} \,\right] \begin{pmatrix} \alpha \\ y \end{pmatrix} = 0. \tag{31}$$

When the $(k_R - m)$-dimensional space of true pairs (α, y) exhausts the kernel of $[(B_1)_{R,J}^\top \mid P_{J,:}]$, Gaussian elimination recovers (α, y) and reads off $z_h^{(1)} = y_h$ from any kernel vector with $(\alpha^\top P_{R,:})_h = 1$.

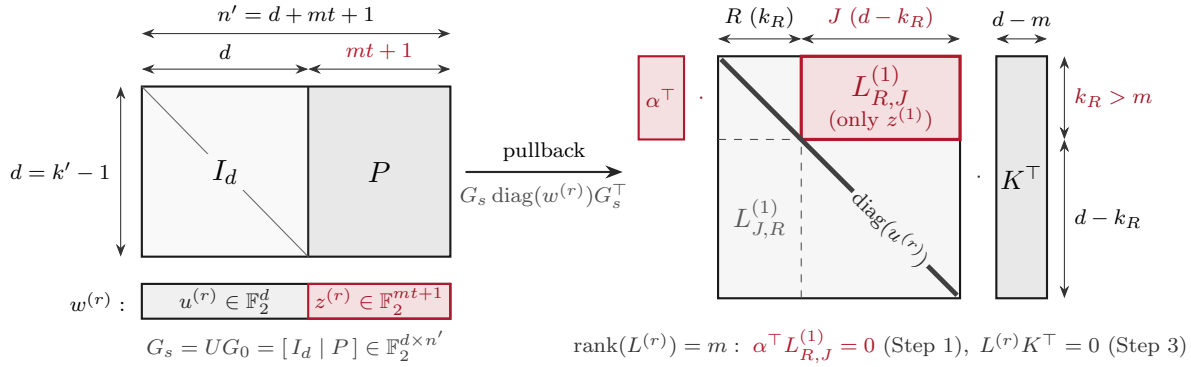

Figure 2: Systematic coordinate decomposition $G_s = [I_d \mid P]$ and block structure of $L^{(r)} = B_r + P \operatorname{diag}(z^{(r)}) P^\top + \operatorname{diag}(u^{(r)}) \in \mathbb{F}_2^{d \times d}$. Trapping all d pivot weights $u^{(r)} \in \mathbb{F}_2^d$ on the main diagonal leaves the $k_R \times (d - k_R)$ off-diagonal block $L^{(1)}_{R,J} = (B_1)_{R,J} + P_{R,:} \operatorname{diag}(z^{(1)}) P^\top_{J,:}$ (highlighted) dependent solely on the $mt + 1$ non-pivot weights $z^{(1)}$. Step 1 recovers $z^{(1)}$ from the left kernel $\alpha^\top L^{(1)}_{R,J} = 0$; Step 2 completes $u^{(1)}$; Step 3 recovers $w^{(2)}, \ldots, w^{(m)}$ against the shared right kernel $K^\top \in \mathbb{F}_2^{d \times (d-m)}$.

2. **Step 2: Sliding Off-Diagonal Kernel for Pivot Diagonal $u^{(1)}$.** With $z^{(1)}$ known, $\mathrm{Off}_1 := B_1 + P \operatorname{diag}(z^{(1)}) P^\top$ is known and equals $L^{(1)}$ off the diagonal. For each index $j \in \{1, \ldots, d\}$, choosing m rows $i_1, \ldots, i_m \neq j$, setting $R = \{j, i_1, \ldots, i_m\}$, and finding the 1-dimensional left kernel $(1, \alpha_1, \ldots, \alpha_m)$ of $(\mathrm{Off}_1)_{R,R^c}$ determines the diagonal entry via column j:
$$L^{(1)}_{j,j} = \sum_{s=1}^{m} \alpha_s (\mathrm{Off}_1)_{i_s,j}, \qquad u^{(1)}_j = L^{(1)}_{j,j} + (\mathrm{Off}_1)_{j,j}. \tag{32}$$

3. **Step 3: Shared-Kernel Multi-RHS Solve for $r = 2, \ldots, m$.** From the completed matrix $L^{(1)}$, we compute its $(d - m) \times d$ right kernel matrix K ($L^{(1)} K^\top = 0$). Because all m matrices $L^{(1)}, \ldots, L^{(m)}$ share the same m-dimensional column space, every $L^{(r)}$ ($r = 2, \ldots, m$) annihilates $K^\top$:
$$L^{(r)} K^\top = 0 \iff \left(G_s \operatorname{diag}(w^{(r)}) G_s^\top \right) K^\top = B_r K^\top \qquad (r = 2, \ldots, m). \tag{33}$$
Because the coefficient operator $w \mapsto (G_s \operatorname{diag}(w) G_s^\top) K^\top$ is independent of r, a single matrix factorization over $\mathbb{F}_2$ recovers all remaining weight vectors $w^{(2)}, \ldots, w^{(m)}$ simultaneously.

Field Alignment, Polynomial Square Root, and $O(n)$ Support Extension. Choosing an invertible $m \times m$ principal block $L^{(1)}_{I,I}$ and forming $A_r = (L^{(1)}_{I,I})^{-1} L^{(r)}_{I,I}$ yields an $\mathbb{F}_2$-subalgebra isomorphic to $\mathbb{F}_{2^m}$ [33, 38]; matching an irreducible characteristic polynomial with $f(x)$ aligns $(w^{(1)}_u, \ldots, w^{(m)}_u)$ with $\mathbb{F}_q$ and recovers $z_u = \sqrt{z_u^2}$ for all $u \in A$. A single harvested weight-$(2t+1)$ codeword $c \in \mathcal{C}_A$ then gives $\bar{g}(X) = \sqrt{\sigma'_S(X)}$ by Theorem 1, and Theorem 2(4) extends L_A to all $n - n'$ remaining columns via $z_j = v_{1,j}/v_{0,j}$ [22] in $O(n)$ field divisions.

4.3 End-to-End Walkthrough on `TII-253`

We now trace the complete Route A pipeline from an algebraic and matrix perspective in terms of the Goppa parameters $(n, m, t, k = n - mt)$ and shortened parameters $(n', k' = n' - mt, d = k' - 1)$. We illustrate every stage on `TII-253` $((n, m, t, k) = (214, 8, 9, 142)$, codimension $mt = 72$, minimum odd weight $w = 2t + 1 = 19)$. As shown in Figure 3(a), rather than solving for the secret support directly in the ambient spaces along the bottom row, Route A follows a zigzag

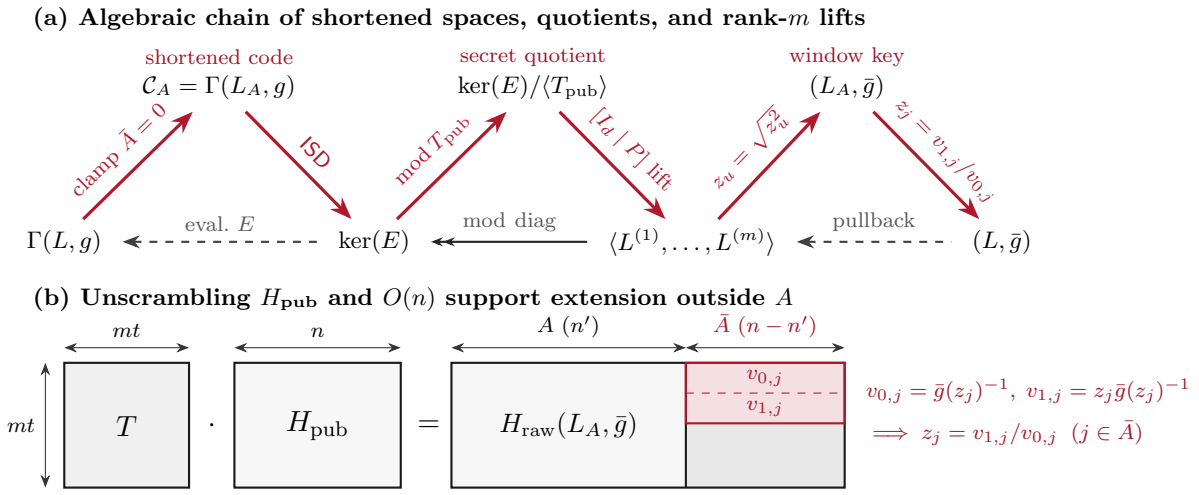

Figure 3: Algebraic and matrix overview of Route A for a binary Goppa code of parameters $(n, m, t, k = n - mt)$. (a) Horizontal black arrows indicate structural evaluation, diagonal-quotient, and pullback maps along the bottom row, while the solid red zigzag path traces the algorithmic steps through the shortened and quotient spaces on the top row. (b) Once $(L_A, \bar{g})$ is known on the n'-column window A, matching $H_{\text{raw}}(L_A, \bar{g})$ on A determines $T \in \mathrm{GL}_{mt}(\mathbb{F}_2)$ and unscrambles TH_{pub} so that the top two m-bit field slices of every column $j \in \bar{A}$ yield $z_j = v_{1,j}/v_{0,j}$.

path (in red) through shortened and quotient spaces along the top row before lifting back to the full private key $(L, \bar{g})$.

1. Primal Shortening and Multi-Target ISD ($H_{\text{pub}} \to \mathcal{C}_A$). Given the public parity-check matrix $H_{\text{pub}} \in \mathbb{F}_2^{mt\times n}$ of a Goppa code $\Gamma(L, g)$ with codimension $mt = n - k$, interpolating the quadratic identity $\sum_{u<v} c_u c_v F(z_u z_v) = F(\bar{g}_{t-1}^2)$ directly over the full code of dimension k would require $F(k-1) = \Theta(k^2)$ monomials and $O(k^6)$ dense linear algebra. Instead, we partition the n columns into an active window A of width $n' = mt + k'$ and a clamped complement $\bar{A}$ of width $n - n'$ (Figure 1, left). Clamping $c_{\bar{A}} = 0$ projects the search onto the shortened code $\mathcal{C}_A = \ker(H_A) = \Gamma(L_A, g) \subset \mathbb{F}_2^{n'}$ of dimension $k' = n' - mt \ll k$, preserving $g(X)$ while leaving the per-check ISD yield $\binom{mt}{w-p}2^{-mt}$ unchanged by Proposition 1. In each ISD round, reducing $H_A\pi$ to $[I_{mt} \mid Q]$ ($Q \in \mathbb{F}_2^{mt\times k'}$, Figure 1, right) collides non-pivot vectors of weight $p = p_1 + p_2$ against pivot syndromes of weight $w - p$ to harvest $W \geq W_{\text{req}}(n', m, t)$ codewords of minimum odd weight $w = 2t+1$. *On `TII-253` ($n = 214, m = 8, t = 9, mt = 72, k = 142, w = 19$):* shortening to $n' = 187$ ($n - n' = 27$ clamped columns) reduces the code dimension from $k = 142$ to $k' = 115$ ($d = k' - 1 = 114$), compressing the quadratic monomial count by **1.53**× from $F(141) = 10{,}012$ to $F(114) = \mathbf{6{,}556}$ (and up to 2.54× across the $(8, 9)$ family at $n = 255$), and Mode 3 ISD ($p = 3$ on $Q \in \mathbb{F}_2^{72\times 115}$) harvests $W = 6{,}643 \geq W_{\text{req}} = 6{,}547$ weight-19 codewords across 64 physical Xeon cores in **0.43** s.

2. Affine Hyperplane and Public Polarization Quotient ($\mathcal{C}_A \to \langle B_1, \ldots, B_m\rangle$). Because every harvested codeword has odd weight $w = 2t + 1$, it lies on the affine hyperplane $\ell \cdot x = 1$ in $\mathbb{F}_2^{k'}$, parameterized by $d = k' - 1 = n' - mt - 1$ coordinates $y \in \mathbb{F}_2^d$ via $c(y) = c_0 + y^\top G_0$ ($G_0 \in \mathbb{F}_2^{d\times n'}$, Figure 3(a)). Evaluating the $F(d) = 1 + d + \binom{d}{2}$ quadratic monomials in y across W harvested codewords produces the evaluation matrix $E \in \mathbb{F}_2^{W\times F(d)}$. By Theorem 2, when $t \neq 5$ the right kernel $\ker(E)$ has dimension $m + 1$, spanned by the m secret bit-slice relations and the public combinatorial relation $\sum_{u<v} c_u c_v = \binom{2t+1}{2} \equiv t \pmod 2$. Quotienting $\ker(E)$ by

its public polarization $T_{\text{pub}} = G_0(J_{n'} + I_{n'})G_0^\top$ isolates an m-dimensional space of symmetric off-diagonal bilinear forms $\langle B_1, \ldots, B_m \rangle \subset \mathbb{F}_2^{d\times d}$. *On* `TII-253`: eliminating $E \in \mathbb{F}_2^{6643\times 6556}$ yields an $(m+1) = 9$-dimensional kernel, and quotienting out $T_{\text{pub}} \in \mathbb{F}_2^{114\times 114}$ leaves $m = 8$ basis matrices $B_1, \ldots, B_8 \in \mathbb{F}_2^{114\times 114}$.

3. Systematic Coordinates and Off-Diagonal Linear Lift ($\langle B_r \rangle \to L_A$). Each basis matrix B_r differs from a rank-m multiplication pullback $L^{(r)} = G_0 \widetilde{M}_{F_r} G_0^\top$ by an unknown diagonal correction $G_0 \operatorname{diag}(w^{(r)}) G_0^\top$, where $(w^{(r)})_u = F_r(z_u^2)$ $(u \in A)$. Transforming $G_0 \in \mathbb{F}_2^{d\times n'}$ into systematic form $G_s = UG_0 = [I_d \mid P]$ $(P \in \mathbb{F}_2^{d\times(mt+1)}$, Figure 2, left) partitions $w^{(r)} = (u^{(r)}, z^{(r)})$ into d pivot weights $u^{(r)} \in \mathbb{F}_2^d$ and $mt+1$ non-pivot weights $z^{(r)} \in \mathbb{F}_2^{mt+1}$, yielding

$$L^{(r)} = B_r + P \operatorname{diag}(z^{(r)}) P^\top + \operatorname{diag}(u^{(r)}) \in \mathbb{F}_2^{d\times d}, \qquad \operatorname{rank}_{\mathbb{F}_2}(L^{(r)}) = m.$$

As depicted in Figure 2 (right), $\operatorname{diag}(u^{(r)})$ is trapped on the main diagonal, so for any partition $\{1, \ldots, d\} = R \sqcup J$ with $|R| = k_R > m$ and $|J| = d - k_R$, the $k_R \times (d - k_R)$ off-diagonal block $L^{(1)}_{R,J} = (B_1)_{R,J} + P_{R,:} \operatorname{diag}(z^{(1)}) P^\top_{J,:}$ depends solely on $z^{(1)}$. Its $(k_R - m)$-dimensional left kernel $\alpha^\top L^{(1)}_{R,J} = 0$ linearizes into $[(B_1)^\top_{R,J} \mid P_{J,:}](\alpha; y) = 0$ (Step 1, recovering $z^{(1)}$), after which sliding $(m+1)$-row off-diagonal kernels determine $u^{(1)}$ (Step 2) and the $(d-m)$-dimensional right kernel $K^\top$ of $L^{(1)}$ solves $L^{(r)} K^\top = 0$ for all $r = 2, \ldots, m$ simultaneously (Step 3). Aligning the $m \times m$ commuting algebra with $\mathbb{F}_q = \mathbb{F}_2[x]/(f(x))$ and taking square roots recovers $z_u = \sqrt{z_u^2}$ for all $u \in A$. *On* `TII-253`: with $G_s = [I_{114} \mid P]$ $(P \in \mathbb{F}_2^{114\times 73})$ and $k_R = 30$, Step 1 solves the 30×84 off-diagonal block $L^{(1)}_{R,J}$ to recover the $mt + 1 = 73$ non-pivot weights $z^{(1)} \in \mathbb{F}_2^{73}$, Step 2 recovers the $d = 114$ pivot weights $u^{(1)} \in \mathbb{F}_2^{114}$, and Step 3 solves against $K^\top \in \mathbb{F}_2^{114\times 106}$ to recover $w^{(2)}, \ldots, w^{(8)} \in \mathbb{F}_2^{187}$ and hence $L_A \in \mathbb{F}_{2^8}^{187}$.

4. Locator Square Root and Unscrambled Support Extension ($L_A \to (L, \bar{g})$). From a single harvested codeword $c \in \mathcal{C}_A$ with support $S \subset A$ $(|S| = 2t + 1)$, Theorem 1 recovers the monic Goppa polynomial $\bar{g}(X) = \sqrt{\sigma'_S(X)}$ in $\mathbb{F}_q[X]$. Finally, choosing an invertible mt-column block $P_0 \subset A$ and computing $T = H_{\text{raw}}(L_A, \bar{g})_{:,P_0} (H_A)^{-1}_{:,P_0} \in \mathrm{GL}_{mt}(\mathbb{F}_2)$ unscrambles the full public parity-check matrix $\widetilde{H} = TH_{\text{pub}} = H_{\text{raw}}(L, \bar{g}) \in \mathbb{F}_2^{mt\times n}$ (Figure 3(b)). Reading the first two m-bit field slices $v_{0,j} = 1/\bar{g}(z_j)$ and $v_{1,j} = z_j/\bar{g}(z_j)$ of each external column $j \in \bar{A}$ immediately recovers the remaining $n - n'$ support elements via $z_j = v_{1,j}/v_{0,j}$ [22]. *On* `TII-253`: inverting a 72×72 block $P_0 \subset A$ unscrambles $\widetilde{H} \in \mathbb{F}_2^{72\times 214}$, and $n - n' = 27$ field divisions in $\mathbb{F}_{2^8}$ recover the remaining 27 support coordinates, completing algebraic recovery and RREF verification in **1.37** s (**1.80** s total wall time, 0.032 ch, 242 MB RAM).

4.4 Complementary Route B on High-Rate Instances

For the single high-rate parameter family $m = 10, t = 11, mt = 110, n \in [1008, 1020]$ $(k \in [898, 910]$, rate $R \approx 0.891 > 1 - 2/t)$, primal shortening in Route A would require a window $n' \approx 443$ $(w = 23)$. Instead, the dual code admits a clean degree-3 $(p_{\mathsf{HOV}} = 3)$ higher-order vanishing (HOV) kernel at dual shortening $s^* = 36$ $(\kappa = mt - s^* = 74, \rho = mp^2_{\mathsf{HOV}} = 90)$ [19, 18, 33]. For these five challenges (`TII-172, 187, 240, 246, 252`), we deploy an optimized HOVER engine (Route B; Appendix A): combining Frobenius rank invariance [33] with the known span dimension $\rho = 90$ replaces the 10-matrix catalecticant rank sum over $\mathbb{F}_{2^{10}}$ by a single rank check $\operatorname{rank} \mathrm{Cat}(h_0) = 9$, incremental commutator intersection stabilizes the algebra E_β from 1–2 right-kernel blocks instead of all 650 blocks, and Segre–Veronese fiber extraction on 450 truncated coordinates is completed via closed-form support extension (25). Route B solves all five $(10, 11)$ instances in 89.10–90.10 s on 1 physical CPU core (0.025 ch), achieving two first-blood recoveries (`TII-172, 187`) and a 4.17× wall-time / 50.0× core-hour speedup over the 12-core reference HOVER solver on `TII-252` [33].

5 Feasibility Boundary and Experimental Verification

5.1 Parameter Regimes and the Classic McEliece Boundary

Across the 170 official Track-2 challenges with $\lambda_{\mathrm{TII}} \geq 83$, the word-supply model in Proposition 1 partitions the catalog into three regimes:

1. **Solved Regime (94 Instances: 89 via Route A, 5 via Route B):** Whenever $n \geq n'_{\min}(m,t)$ and $mt \leq 126$, the code supplies enough weight-$(2t+1)$ codewords to determine the $F(d)$ quadratic features, and our framework recovers the private key (L, g).

2. **Information-Theoretically Underdetermined Regime (60 Challenges with $n < n'_{\min}$):** In 60 challenges with short lengths n relative to mt (e.g., `TII-201` with $m = 8, t = 20, n = 253, k = 93$, where $W_{\mathrm{req}} = 4{,}270$ evaluation rows are required on $d = 92$ coordinates but the full code has only $\binom{253}{41}2^{-160} \approx 0.224$ expected weight-41 codewords), the full code does not contain enough minimum-weight codewords to determine the quadratic system regardless of available compute.

3. **Compute-Limited Regime (16 Challenges with $mt \in [120, 189]$) and Classic McEliece:** In the remaining 16 challenges with $n \geq n'_{\min}$ and large $mt \in [120, 189]$ ($w \in [25, 43]$), the codeword supply exists in principle, but the per-iteration ISD yield (20) scales as $\binom{k'}{p}\binom{mt}{w-p}2^{-mt}$, requiring 10^4–10^8+ core-hours. For Classic McEliece standard parameters ($m \in \{12, 13\}$, $t \geq 64$, $mt \geq 768$, $w \geq 129$, $n \in [3488, 8192]$), Route A is doubly infeasible: for `mceliece348864` ($m = 12, t = 64, mt = 768, w = 129$) at the algebraic threshold $n'_{\min} = 793$, the expected count of weight-129 codewords is $\binom{793}{129}2^{-768} < 2^{-264}$ (and $< 2^{-20}$ even at full length n for all $m = 13$ Classic McEliece parameter sets), while at full length $n = 3488$ ($\log_2 \binom{3488}{129} - 768 \approx 23.47$), multi-target ISD for a weight-129 codeword exceeds 2^{240} operations [8, 3] (and even single-target weight-64 syndrome decoding requires $2^{150.6}$ bit operations in `CryptAttackTester` [5]).

5.2 Verification and Compute Accounting on 94 Challenges

We evaluated Route A and Route B across the official TII McEliece Track-2 catalog and verified all 94 recovered keys:

- **Official SageMath Verification (94/94 Verified):** Every recovered key is formatted in TII's official two-line (g, L) representation and passes a complete rerun of TII's official SageMath verification script under SageMath 10.9 against the official public-key files (94/94 verified), with monic $\mathbb{F}_q$-irreducible $g(X)$ of degree t across all 94 keys.

- **Physical Core-Hour Accounting:** We count allocated physical CPU cores (excluding hyperthreads) multiplied by end-to-end wall-clock runtime across both ISD sampling and algebraic lifting. Under this metric, the 92 single-node challenges consume 177.82 physical core-hours combined (87 via Route A on a 64-physical-core Dual Intel Xeon Gold 6338 server, and 5 via Route B on 1 physical core), while the two pure-CPU cluster challenges `TII-228` ($500, 9, 14, mt = 126$) and `TII-250` ($1010, 10, 12, mt = 120$) consume 2,463.3 and 66,722.1 physical core-hours (57.1 min and 9.20 h wall time). Table 4 summarizes all 94 solved challenges by parameter family.

A Frobenius-Reduced HOVER (Route B)

This appendix details the mathematical formulation, algorithmic optimizations, and per-instance experimental results of Route B on all five official TII Track-2 challenges in the high-rate family $m = 10, t = 11, mt = 110, n \in [1008, 1020]$ (`TII-172, 187, 240, 246, 252`).

Table 4: Summary of all 94 solved TII Track-2 key-recovery challenges grouped by Goppa parameter family (m, t).

Family	Route	Solved	Window	Med. Time	Med. Compute	Total Compute	RAM	Challenge IDs
$(7,5)$	A (35)	3 (0)	$n' = 120$–122	10.9 s	0.194 ch	3.72 ch	237 MB	86, 90, 94
$(7,7)$	A (49)	6 (6)	$n' = 112$	0.39 s	0.007 ch	0.04 ch	231 MB	95, 103, 107, 111, 127, 133
$(7,8)$	A (56)	4 (4)	$n' = 120$–124	3.53 s	0.063 ch	0.24 ch	233 MB	87, 97, 102, 114
$(7,9)$	A (63)	1 (1)	$n' = 127$	1.76 s	0.031 ch	0.03 ch	228 MB	88
$(8,5)$	A (40)	2 (0)	$n' = 143$	1.53 s	0.027 ch	0.05 ch	286 MB	83, 89
$(8,6)$	A (48)	2 (0)	$n' = 152$	0.96 s	0.017 ch	0.03 ch	340 MB	84, 91
$(8,7)$	A (56)	7 (0)	$n' = 163$	1.06 s	0.019 ch	0.12 ch	340 MB	92, 110, 160, 171, 178, 212, 220
$(8,8)$	A (64)	10 (7)	$n' = 175$	1.32 s	0.023 ch	0.24 ch	340 MB	85, 93, 100, 112, 118, 123, 179, 186, 233, 238
$(8,9)$	A (72)	7 (6)	$n' = 187$	1.38 s	0.024 ch	0.18 ch	252 MB	101, 120, 126, 146, 216, 222, 253
$(8,10)$	A (80)	10 (10)	$n' = 199$	2.02 s	0.036 ch	0.36 ch	255 MB	109, 122, 128, 139, 144, 149, 221, 224, 227, 230
$(8,11)$	A (88)	4 (4)	$n' = 212$	3.75 s	0.067 ch	0.29 ch	259 MB	130, 152, 157, 206
$(8,12)$	A (96)	3 (2)	$n' = 223$–224	28.8 s	0.511 ch	1.22 ch	313 MB	183, 203, 254
$(8,13)$	A (104)	1 (1)	$n' = 233$	332 s	5.90 ch	5.90 ch	254 MB	232
$(8,14)$	A (112)	3 (3)	$n' = 246$–250	82.7 s	1.47 ch	10.77 ch	263 MB	132, 165, 190
$(9,7)$	A (63)	6 (0)	$n' = 237$	5.50 s	0.098 ch	0.60 ch	340 MB	119, 125, 166, 210, 244, 248
$(9,8)$	A (72)	6 (0)	$n' = 253$	11.4 s	0.203 ch	1.21 ch	340 MB	96, 105, 113, 134, 164, 200
$(9,9)$	A (81)	4 (0)	$n' = 269$	39.8 s	0.708 ch	2.85 ch	339 MB	129, 189, 209, 223
$(9,10)$	A (90)	4 (3)	$n' = 286$	113 s	2.02 ch	8.73 ch	351 MB	138, 158, 208, 213
$(9,11)$	A (99)	1 (1)	$n' = 303$	408 s	7.25 ch	7.25 ch	373 MB	147
$(9,12)$	A (108)	3 (3)	$n' = 326$	2509 s	44.61 ch	133.84 ch	427 MB	182, 188, 245
$(9,14)$	A (126)	1 (1)	$n' = 361$	57.1 min	2,463 ch	2,463 ch	468 MB	228
$(10,11)$	B (110)	5 (2)	$s^* = 36, \kappa = 74$	89.9 s	0.025 ch	0.12 ch	1.49 GB	172, 187, 240, 246, 252
$(10,12)$	A (120)	1 (1)	$n' = 467$	9.20 h	66,722 ch	66,722 ch	1.28 GB	250
Total	**A+B**	**94 (55)**	—	—	—	**69,363.3** ch	—	92 **single-node:** 177.82 **ch**

Notes: For each family (m, t), 'Route' lists the engine and codimension mt; 'Solved' lists total (and first-blood) solutions; 'Window' gives n' (Route A) or dual shortening s^* (Route B); compute is in physical CPU core-hours (ch).

A.1 Shortened-Dual Degree-3 HOV Formulation

In the length-n dual code $\Gamma(L, g)^\perp \subset \mathbb{F}_2^n$ (spanned by the $mt = 110$ rows of H_{pub}), shortening at $s^* = 36$ coordinates produces a shortened dual code D_{short} of dimension $\kappa = mt - s^* = 74$ and length $n - s^* \in [972, 984]$. Because the rows of $H_{\text{raw}}(L, g)$ are vertical $\mathbb{F}_2$-bit slices of rational evaluations $z_j^i / g(z_j)$ over $\mathbb{F}_{2^{10}}$, requiring square-free cubic ($p_{\mathsf{HOV}} = 3$) forms in $\Lambda^3(\mathbb{F}_2^\kappa)$ (of dimension $\binom{\kappa}{3} = \binom{74}{3} = 64{,}824$) to vanish together with all first partial derivatives on codewords of D_{short} leaves a clean 10-dimensional *higher-order vanishing kernel* [19, 18, 33]:

$$K_{\mathsf{HOV}} = \mathcal{V}_3(D_{\text{short}}) \subset \Lambda^3(\mathbb{F}_2^\kappa), \qquad \dim_{\mathbb{F}_2} K_{\mathsf{HOV}} = m = 10. \tag{34}$$

Each cubic form $h \in K_{\mathsf{HOV}}$ defines, via its partial derivatives (its *first catalecticant*), a linear map $\mathrm{Cat}(h) : \mathbb{F}_2^\kappa \to \Lambda^2(\mathbb{F}_2^\kappa)$ represented by a $\binom{\kappa}{2} \times \kappa = 2{,}701 \times 74$ matrix. Stacked horizontally across an $\mathbb{F}_2$-basis $f_1, \ldots, f_m$ of K_{HOV}, their catalecticants span a row space of dimension $\rho = m p_{\mathsf{HOV}}^2 = 10 \times 9 = 90$, yielding a row-compressed stacked catalecticant $T \in \mathbb{F}_2^{\rho \times \kappa m} = \mathbb{F}_2^{90 \times 740}$.

Saarinen [33] showed that the hidden extension field $\mathbb{F}_{2^m}$ acts on K_{HOV} as an m-dimensional commutative endomorphism algebra $E_\beta \subset \mathbb{F}_2^{m \times m}$ whose induced operator $(I_\kappa \otimes B)$ stabilizes the $n_z = \kappa m - \rho = 740 - 90 = 650$-dimensional right kernel $\ker(T) \subset \mathbb{F}_2^{\kappa m}$. However, the reference HOVER implementation [33] incurs three computational bottlenecks:

1. **Full Commutator Stacking:** To recover E_β, it stacks all $n_z = 650$ right-kernel invariance blocks $T(I_\kappa \otimes B)z = 0$ of size 90×100 (58,500 rows) before solving;

2. **All-m Catalecticant Rank Evaluations Over $\mathbb{F}_{2^{10}}$:** Although Frobenius transport equates catalecticant ranks across a Galois orbit [33], the reference selector evaluates all $m = 10$ catalecticant matrices $\mathrm{Cat}(h_r) \in \mathbb{F}_{2^{10}}^{2701\times 74}$ to check $\sum_{r=0}^{m-1} \operatorname{rank} \mathrm{Cat}(h_r) = \rho = 90$;

3. **Full-Length Segre–Veronese Fiber Extraction:** Following single-direction selection, it computes Segre–Veronese tangent spaces and quadratic fibers across all $n - s^* \approx 972$–984 dual-unshortened coordinates.

A.2 Single-Direction Rank Reduction and Incremental Extraction

Route B eliminates these three bottlenecks via Proposition 3:

Proposition 3 (Single-Direction Selector Reduction and Incremental E_β Recovery)**.** *Let $h_0, h_1, \ldots, h_{m-1} \in K_{\mathsf{HOV}} \otimes_{\mathbb{F}_2} \mathbb{F}_{2^m}$ be the m eigenlines of a primitive generator $B \in E_\beta$, ordered so that their eigenvalues form the Frobenius conjugate orbit $\lambda, \lambda^2, \ldots, \lambda^{2^{m-1}} \in \mathbb{F}_{2^m}$.*

1. ***Single-Matrix Catalecticant Selector:*** *Because the basis $f_1, \ldots, f_m$ of K_{HOV} is defined over $\mathbb{F}_2$, Frobenius transport [33] maps each eigenline h_r to h_{r+1} up to a nonzero scalar $\mu_r \in \mathbb{F}_{2^m}^\times$:*

$$h_{r+1} = \mu_r\, h_r^{[2]} \implies \mathrm{Cat}(h_{r+1}) = \mu_r\, \mathrm{Cat}(h_r)^{[2]} \implies \operatorname{rank} \mathrm{Cat}(h_{r+1}) = \operatorname{rank} \mathrm{Cat}(h_r). \tag{35}$$

   Combining (35) with the already-computed row-space dimension $\rho = 90$ reduces the m-matrix HOVER *selector condition $\sum_{r=0}^{m-1} \operatorname{rank} \mathrm{Cat}(h_r) = \rho$ to the single-matrix rank test $m \cdot \operatorname{rank} \mathrm{Cat}(h_0) = \rho$ (i.e., $\operatorname{rank} \mathrm{Cat}(h_0) = p_{\mathsf{HOV}}^2 = 9$), eliminating $m - 1 = 9$ of the 10 catalecticant constructions and Gaussian eliminations over $\mathbb{F}_{2^{10}}$.*

2. ***Rank-Checked Incremental Stabilization of E_β:*** *In the clean kernel regime, the true m-dimensional algebra E_β lies in the intersection of invariance constraints $T(I_\kappa \otimes B)z_b = 0$ for every prefix of right-kernel vectors $z_1, \ldots, z_b \in \ker(T)$. Intersecting blocks incrementally drops the candidate subspace to the target dimension $\dim_{\mathbb{F}_2} E_\beta = m = 10$ after 1 or 2 blocks across all five $(10, 11)$ instances, cutting Stage 2 runtime to* **0.18** *s.*

3. ***Truncated* 450*-Coordinate Fiber Labeling and Extension:*** *Restricting single-direction Segre–Veronese fiber extraction [33] to 450 coordinates, verifying multiplier dimensions and $mt = 110$ independent compatible columns P_0, recovering $\bar{g}(X)$ and $T \in \mathrm{GL}_{110}(\mathbb{F}_2)$, and applying closed-form ratio extension [22, 33] ($z_j = v_{1,j}/v_{0,j}$, Theorem 2(4)) recovers all remaining coordinates in $O(n)$ field divisions.*

Proof. Write the $\mathbb{F}_2$-basis of K_{HOV} as $f_1, \ldots, f_m$ and let $v^{(r)} = (v_1^{(r)}, \ldots, v_m^{(r)})^\top \in \mathbb{F}_{2^m}^m$ be the r-th eigenvector of $B^\top$ over $\mathbb{F}_{2^m}$ defining $h_r = \sum_{i=1}^m v_i^{(r)} f_i$. Because $B \in \mathbb{F}_2^{m\times m}$ has entries in $\mathbb{F}_2$ ($B^{[2]} = B$), applying the Frobenius automorphism $x \mapsto x^2$ entrywise to $B^\top v^{(r)} = \lambda^{2^r} v^{(r)}$ yields

$$B^\top (v^{(r)})^{[2]} = (B^\top v^{(r)})^{[2]} = (\lambda^{2^r})^2 (v^{(r)})^{[2]} = \lambda^{2^{r+1}} (v^{(r)})^{[2]}.$$

Since the m Frobenius conjugates λ^{2^r} of a degree-m primitive element are pairwise distinct, each eigenspace is 1-dimensional, so $v^{(r+1)} = \mu_r (v^{(r)})^{[2]}$ for some $\mu_r \in \mathbb{F}_{2^m}^\times$. Because the catalecticant polarization map is $\mathbb{F}_2$-linear, $\mathrm{Cat}(h_{r+1}) = \mu_r\, \mathrm{Cat}(h_r)^{[2]}$. Since field automorphisms and nonzero scalar multiplication preserve matrix rank, $\operatorname{rank} \mathrm{Cat}(h_{r+1}) = \operatorname{rank} \mathrm{Cat}(h_r)$, establishing (35). Parts 2 and 3 rely on explicit dimension and compatibility verification at runtime before accepting the extended key. □

A.3 Per-Instance Experimental Results

Table 5 reports the per-instance experimental results for all five $(m, t) = (10, 11)$ challenges solved by Route B on 1 physical CPU core. On `TII-252` (90.03 s total), the runtime comprises 58.2 s for Stage 1 (bit-sliced degree-3 HOV kernel K_{HOV}), 0.18 s for Stage 2 (1–2 block incremental E_β stabilization), 1.85 s for Stage 3 (single-direction Frobenius rank test $\operatorname{rank} \mathrm{Cat}(h_0) = 9$), and 29.8 s for Stages 4–5 (450-coordinate fiber extraction and $O(n)$ support extension).

Table 5: Per-instance key-recovery performance of Route B on all five $(m, t) = (10, 11)$ TII Track-2 challenges.

ID	Parameters	Prior Time	Prior Compute	Our Time	Our Compute	RAM	Speedup / Status
172	$(1020, 10, 11, 910)$	*No prior published solution*		**89.85** s	**0.025** ch	1.51 GB	**First-Blood**
187	$(1018, 10, 11, 908)$	*No prior published solution*		**90.10** s	**0.025** ch	1.50 GB	**First-Blood**
240	$(1010, 10, 11, 900)$	186 s [33]	0.620 ch (12 cores)	**89.10** s	**0.025** ch	1.49 GB	**2.09**× time / **24.8**× ch
246	$(1009, 10, 11, 899)$	191 s [33]	0.637 ch (12 cores)	**89.13** s	**0.025** ch	1.48 GB	**2.14**× time / **25.5**× ch
252	$(1008, 10, 11, 898)$	375 s [33]	1.250 ch (12 cores)	**90.03** s	**0.025** ch	1.45 GB	**4.17**× time / **50.0**× ch

Notes: All five instances use dual shortening $s^* = 36$ ($\kappa = 74, \rho = 90$) on 1 physical CPU core (ch = physical core-hours).

Use of AI. The cryptanalytic results and artifacts were initially developed through an autonomous agent system powered by a MiniMax internal model. We validated the key technical claims and refined the presentation of the manuscript.

References

[1] M. Bardet, M. Bros, D. Cabarcas, P. Gaborit, R. Perlner, D. Smith-Tone, J.-P. Tillich, and J. Verbel, "Improvements of algebraic attacks for solving the rank decoding and MinRank problems," in *Advances in Cryptology – ASIACRYPT 2020*, Part I, Lecture Notes in Computer Science, vol. 12491, pp. 507–536, Springer, 2020.

[2] M. Bardet, A. Lemoine, and J.-P. Tillich, "An attack on the CFS scheme and on TII McEliece challenges," in *Advances in Cryptology – CRYPTO 2026*, Part IV, Lecture Notes in Computer Science, vol. 16803, pp. 380–412, Springer, 2026. Full version: IACR Cryptology ePrint Archive, Report 2026/430, https://eprint.iacr.org/2026/430.

[3] A. Becker, A. Joux, A. May, and A. Meurer, "Decoding random binary linear codes in $2^{n/20}$: How $1 + 1 = 0$ improves information set decoding," in *Advances in Cryptology – EUROCRYPT 2012*, Lecture Notes in Computer Science, vol. 7237, pp. 520–536, Springer, 2012.

[4] E. Berlekamp, R. McEliece, and H. van Tilborg, "On the inherent intractability of certain coding problems," *IEEE Transactions on Information Theory*, vol. 24, no. 3, pp. 384–386, 1978.

[5] D. J. Bernstein and T. Chou, "CryptAttackTester: High-assurance attack analysis," in *Advances in Cryptology – CRYPTO 2024*, Part VI, Lecture Notes in Computer Science, vol. 14925, pp. 141–182, Springer, 2024.

[6] D. J. Bernstein, T. Lange, and C. Peters, "Smaller decoding exponents: ball-collision decoding," in *Advances in Cryptology – CRYPTO 2011*, Lecture Notes in Computer Science, vol. 6841, pp. 743–760, Springer, 2011.

[7] P. Briaud, A. Lemoine, H. Randriambololona, and J.-P. Tillich, "A Heuristic Subexponential Attack on the McEliece Cryptosystem," IACR Cryptology ePrint Archive, Report 2026/1232, 2026 (to appear in *ASIACRYPT 2026*). https://eprint.iacr.org/2026/1232.

[8] A. Canteaut and F. Chabaud, "A new algorithm for finding minimum-weight words in a linear code: Application to McEliece's cryptosystem and to narrow-sense BCH codes of length 511," *IEEE Transactions on Information Theory*, vol. 44, no. 1, pp. 367–378, 1998.

[9] K. Carrier, T. Debris-Alazard, C. Meyer-Hilfiger, and J.-P. Tillich, "Reduction from sparse LPN to LPN, Dual Attack 3.0," in *Advances in Cryptology – EUROCRYPT 2024*, Part VII, Lecture Notes in Computer Science, vol. 14657, pp. 286–315, Springer, 2024.

[10] A. Couvreur, P. Gaborit, V. Gauthier-Umaña, A. Otmani, and J.-P. Tillich, "Distinguisher-based attacks on public-key cryptosystems using Reed–Solomon codes," *Designs, Codes and Cryptography*, vol. 73, no. 2, pp. 641–666, 2014.

[11] A. Couvreur, R. Mora, and J.-P. Tillich, "A new approach based on quadratic forms to attack the McEliece cryptosystem," in *Advances in Cryptology – ASIACRYPT 2023*, Part IV, Lecture Notes in Computer Science, vol. 14441, pp. 3–38, Springer, 2023.

[12] M. Cueto Noval, S.-P. Merz, P. Stähli, and A. Ünal, "On the soundness of algebraic attacks against code-based assumptions," in *Advances in Cryptology – EUROCRYPT 2025*, Part VI, Lecture Notes in Computer Science, vol. 15606, pp. 385–415, Springer, 2025.

[13] L. Ducas, A. Esser, S. Etinski, and E. Kirshanova, "Asymptotics and improvements of sieving for codes," in *Advances in Cryptology – EUROCRYPT 2024*, Part VII, Lecture Notes in Computer Science, vol. 14657, pp. 151–180, Springer, 2024.

[14] I. Dumer, "On minimum distance decoding of linear codes," in *Proc. 5th Joint Soviet–Swedish Int. Workshop on Information Theory*, pp. 50–52, 1991.

[15] J.-C. Faugère, V. Gauthier-Umaña, A. Otmani, L. Perret, and J.-P. Tillich, "A distinguisher for high-rate McEliece cryptosystems," *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6830–6844, 2013.

[16] M. Finiasz, "Words of minimal weight and weight distribution of binary Goppa codes," in *Proc. 2003 IEEE International Symposium on Information Theory (ISIT 2003)*, p. 70, IEEE, 2003.

[17] A. Ghoshal, Y. Ishai, A. Jain, and N. Sun, "Quasipolynomial Cryptanalysis of the McEliece Cryptosystem (or: PIR Meets McEliece)," IACR Cryptology ePrint Archive, Report 2026/1630, 2026. https://eprint.iacr.org/2026/1630.

[18] T. Hemmert, "Key recovery for the McEliece cryptosystem using higher-order vanishing," IACR Cryptology ePrint Archive, Report 2026/1339, 2026. https://eprint.iacr.org/2026/1339.

[19] T. Hemmert and A. Wiemers, "Distinguishing Goppa codes using higher-order vanishing," in *Advances in Cryptology – CRYPTO 2026*, Part IV, Lecture Notes in Computer Science, vol. 16803, pp. 191–223, Springer, 2026.

[20] A. Kipnis and A. Shamir, "Cryptanalysis of the HFE public key cryptosystem by relinearization," in *Advances in Cryptology – CRYPTO '99*, Lecture Notes in Computer Science, vol. 1666, pp. 19–30, Springer, 1999.

[21] E. Kirshanova and A. Esser, "TII McEliece Challenges: Scripts and Instances," Technology Innovation Institute (TII), 2023. Challenge portal: https://crowdchallenge.tii.ae/mceliece-challenges/; data repository: https://github.com/ElenaKirshanova/tii_decoding_challenge.

[22] E. Kirshanova and A. May, "Breaking Goppa-based McEliece with hints," *Information and Computation*, vol. 293, art. 105045, 2023.

[23] P. J. Lee and E. F. Brickell, "An observation on the security of McEliece's public-key cryptosystem," in *Advances in Cryptology – EUROCRYPT '88*, Lecture Notes in Computer Science, vol. 330, pp. 275–280, Springer, 1988.

[24] J. S. Leon, "A probabilistic algorithm for computing minimum weights of large error-correcting codes," *IEEE Transactions on Information Theory*, vol. 34, no. 5, pp. 1354–1359, 1988.

[25] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*, North-Holland, Amsterdam, 1977, ch. 12.

[26] A. May, A. Meurer, and E. Thomae, "Decoding random linear codes in $\tilde{\mathcal{O}}(2^{0.054n})$," in *Advances in Cryptology – ASIACRYPT 2011*, Lecture Notes in Computer Science, vol. 7073, pp. 107–124, Springer, 2011.

[27] A. May and I. Ozerov, "On computing nearest neighbors with applications to decoding of binary linear codes," in *Advances in Cryptology – EUROCRYPT 2015*, Part I, Lecture Notes in Computer Science, vol. 9056, pp. 203–228, Springer, 2015.

[28] R. J. McEliece, "A public-key cryptosystem based on algebraic coding theory," *DSN Progress Report*, vol. 42-44, pp. 114–116, Jet Propulsion Laboratory, 1978.

[29] H. Niederreiter, "Knapsack-type cryptosystems and algebraic coding theory," *Problems of Control and Information Theory*, vol. 15, no. 2, pp. 159–166, 1986.

[30] L. Panny, "On breaking McEliece keys using brute force," in *Post-Quantum Cryptography – PQCrypto 2026*, Part II, Lecture Notes in Computer Science, vol. 16492, pp. 275–308, Springer, 2026. Full version: IACR Cryptology ePrint Archive, Report 2025/632, https://eprint.iacr.org/2025/632.

[31] N. J. Patterson, "The algebraic decoding of Goppa codes," *IEEE Transactions on Information Theory*, vol. 21, no. 2, pp. 203–207, 1975.

[32] H. Randriambololona, "The syzygy distinguisher," in *Advances in Cryptology – EUROCRYPT 2025*, Part VI, Lecture Notes in Computer Science, vol. 15606, pp. 324–354, Springer, 2025.

[33] M.-J. O. Saarinen, "HOVER: Higher-Order Vanishing Endomorphism Recovery," IACR Cryptology ePrint Archive, Report 2026/1778, 2026 (to appear in *NordSec 2026*). https://eprint.iacr.org/2026/1778.

[34] M.-J. O. Saarinen, "Two-Anchor Holdout/Hermite: Solving the TII-254 McEliece Key Recovery Challenge," IACR Cryptology ePrint Archive, Report 2026/1986, 2026. https://eprint.iacr.org/2026/1986.

[35] M.-J. O. Saarinen, *tii-solved — recovered keys for previously unsolved TII McEliece challenges*, accompanying artifact repository, Results table, accessed November 6, 2026. https://github.com/mjosaarinen/tii-solved.

[36] N. Sendrier, "Decoding one out of many," in *Post-Quantum Cryptography – PQCrypto 2011*, Lecture Notes in Computer Science, vol. 7071, pp. 51–67, Springer, 2011.

[37] J. Stern, "A method for finding codewords of small weight," in *Coding Theory and Applications*, Lecture Notes in Computer Science, vol. 388, pp. 106–113, Springer, 1989.

[38] K. Vedenev, "Extending Distinguishing to Key Recovery for Subfield Subcodes of GRS codes," IACR Cryptology ePrint Archive, Report 2026/1747, 2026. https://eprint.iacr.org/2026/1747.

[39] S. A. Weis, "Improving GIJS Key Recovery for Classic McEliece," IACR Cryptology ePrint Archive, Report 2026/1984, 2026. https://eprint.iacr.org/2026/1984.