Bit Operation Cost of "Holdout" Key-Recovery Attacks Against Classic McEliece

Markku-Juhani O. Saarinen

Tampere University, Finland
markku-juhani.saarinen@tuni.fi

Revision: 15 September 2026

Abstract. We give an itemized conditional arithmetic estimate for "Holdout" key recovery on each Classic McEliece parameter set. This attack uses polynomial relations to recover locators, the hidden field values specifying the code. Shortening restricts codewords to zero on selected positions and then deletes those positions. Assuming the public tests uniquely recover enough compatible locator values, one shortened coordinate set suffices for each parameter set, instead of the four or five in the analysis of Ghoshal, Ishai, Jain, and Sun. We make the linear solves reliable using Eberly's random scaling of rows and columns and scalar Lanczos, an iterative solver over a larger field. Returned solutions are mapped back and checked in the original equations. Charging operations with binary coefficients separately from general field arithmetic gives conditional arithmetic estimates of $2^{126.77}$ bit operations for mceliece348864, $2^{145.22}$ for mceliece460896, $2^{137.48}$ for mceliece6688128, $2^{136.65}$ for mceliece6960119, and $2^{137.48}$ for mceliece8192128. The corresponding working-memory estimates are $2^{51.33}$, $2^{59.10}$, $2^{55.18}$, $2^{54.86}$, and $2^{55.18}$ bits. All five arithmetic estimates lie below the NIST classical-gate reference level for their claimed category; the model excludes address generation and memory traffic and uses budget estimates for some stages.

The estimates depend on structural conditions not yet established at Classic McEliece size. We show how rank and determinant tests guarantee the expected relations, but do not prove that these tests, the required public matrix ranks, or the recovery of unique locator values succeed for the displayed parameter sets. On small generated instances, the local tests sometimes accepted several candidate values. Choices accepted separately need not work together, and a different method nevertheless recovered one of these small keys. Lower conditional estimates for newer routes are discussed separately. These are estimates of arithmetic work and working memory, not elapsed time; no Classic McEliece target key has been recovered.

Note: This is a living document: the estimate, assumptions, and scope will be updated as algorithms and reproducible evidence improve.

Keywords: Classic McEliece · Holdout Attack · Key Recovery · Bit Operations · Sparse Linear Algebra

1 Introduction

Classic McEliece derives its public code from a binary Goppa code [3,9]. Recent algebraic attacks turn publicly computable polynomial relations into candidate support values, called *locators*, and then conditionally into an equivalent decoding key [5,6,7,12]. Their concrete cost is sensitive to three details that an algebraic-operation count can hide: how many shortened coordinate sets must be processed, which matrix entries are binary, and whether the linear-system solver has a quantified failure probability.

Terminology. Write m for the field extension degree, $q = 2^m$ for the field size and t for the degree of the hidden Goppa polynomial. A polynomial relation is a linear dependence among polynomial values and derivatives at public code columns. A *holdout* is a public coordinate deliberately omitted from a set of vanishing constraints so that the resulting relations can reveal local information at that coordinate. A *projective label* is a proposed locator in the projective line $\mathbb{P}^1(\mathbb{F}_q)$, consisting of the q field elements and one point at infinity. The direct route repeatedly tests such labels at held-out coordinates; accepted labels are used to reconstruct the rest of the key, which is then checked against the public code.

A *canonical relation branch* is the expected family of relations associated with one representation of the hidden code. Rank two means that these relations span a two-dimensional vector space. An *affine feasibility test* asks whether a linear system $Ax = b$ has a solution, allowing a nonzero right-hand side b. A matrix is *sparse* when most of its entries are zero.

Current Revision. This paper accounts for those three costing questions for the direct locator-recovery route in the 27 August 2026 revision of Ghoshal et al. [5]. We report one fixed parameter choice for each set, without claiming optimality among all conditional routes. Weis gives lower conditional estimates under different assumptions [13]. Section 6 compares these and other proposed routes, including Vedenev's 6 September working draft [12]. Section B separates the arithmetic ledger from the budget estimates that still need implementation-level justification.

The calculation is conditional in the precise sense of Section 7. In particular, the paper does not assign probabilities to unproved structural conjectures.

Our contributions:

1. a reduction of the conjectured canonical rank-two relation branch to one rank test on pairwise polynomial products and explicit t-by-t determinants that test the required interpolation maps;
2. a rank and capacity proof that one shortened coordinate set suffices for key completion in every Classic McEliece parameter set, conditional on the stated requirements for uniquely recovered, compatible locator values;

3. a reliable method for solving the affine sparse systems over a larger field, including an exact linear map back to the original field and substitution into the original equations;
4. an explicit decomposition into binary operations and general field operations, with a finite estimate for every remaining algebraic step; and
5. a small self-contained artifact, `mccost.py`, that recomputes the dimensions, exact rational reliability bounds, work and state ledgers, and five current-route totals in this paper.

The public artifact is available at `github.com/mjosaarinen/mccost`.

2 The Direct Locator-Recovery System

Let m be the extension degree, $q = 2^m$ the field size, n the code length, t the degree of the secret Goppa polynomial, and k the public-code dimension. The National Institute of Standards and Technology (NIST) post-quantum evaluation criteria use 2^{143}, 2^{207}, and 2^{272} classical gates as reference levels for Categories 1, 3, and 5, respectively [10]. These are reference comparisons for the bit-operation metric used here; they are not simply 2^{128}, 2^{192}, and 2^{256} elementary operations. Our arithmetic tally excludes integer index and address generation, memory traffic, and communication. Its numerical comparison with these reference levels therefore does not certify the gate count of a complete implementation.

Table 1 gives the five fixed parameter choices, or *cells*, from the source analysis. *Shortening* on ℓ coordinates means restricting to codewords that are zero there and then deleting those coordinates. When the deleted generator columns have rank ℓ, as required here, the remaining length and dimension are $(n_\ell, k_\ell) = (n - \ell, k - \ell)$. We call the resulting shortened coordinate set a *chart*, and write d for the degree of the polynomial relations.

A polynomial is *homogeneous* when all of its monomials have the same total degree. A Hermite jet at a point is the collection of a polynomial's value and its Hasse derivatives, the coefficient-based derivatives appropriate in finite characteristic, up to a chosen order; multiplicity s means imposing vanishing for derivative orders below s. The conditions at one point form a *block*. In the profile column, s^a means that a points receive multiplicity s. We use the source's cells without re-optimizing them.

Table 1. Direct locator-recovery cells and NIST classical-gate reference levels [10].

Parameter set	Category / reference	(m, n, t, k)	(ℓ, d)	(n_ℓ, k_ℓ)	Profile
mceliece348864	1 / 2^{143}	$(12, 3488, 64, 2720)$	$(2569, 7)$	$(919, 151)$	$5^{42}6^{877}$
mceliece460896	3 / 2^{207}	$(13, 4608, 96, 3360)$	$(3159, 8)$	$(1449, 201)$	$5^{4}6^{177}7^{1268}$
mceliece6688128	5 / 2^{272}	$(13, 6688, 128, 5024)$	$(4802, 7)$	$(1886, 222)$	$5^{35}6^{1851}$
mceliece6960119	5 / 2^{272}	$(13, 6960, 119, 5413)$	$(5198, 7)$	$(1762, 215)$	$5^{24}6^{1738}$
mceliece8192128	5 / 2^{272}	$(13, 8192, 128, 6528)$	$(6306, 7)$	$(1886, 222)$	$5^{35}6^{1851}$

Let s_j be the multiplicity at position j, fix the holdout multiplicity $s = 5$, and put $D = n_\ell - 2t - 1$. Every profile uses exactly the required number of interpolation conditions after omitting one holdout, with no surplus. This gives the *no-slack* identity [5, Eq. (12)]

$$\sum_{j=1}^{n_\ell} s_j = dD + 1 - t + s. \tag{1}$$

Thus omitting one multiplicity-s holdout leaves exactly $dD + 1 - t$ univariate interpolation conditions. The artifact checks this identity for all five cells.

For a detailed derivation, we use mceliece348864 as the running example. Its parameters are

$$(m, n, t, k) = (12, 3488, 64, 2720). \tag{2}$$

Its fixed cell has $(\ell, d) = (2569, 7)$, $(n_\ell, k_\ell) = (919, 151)$, and profile $5^{42}6^{877}$. The same formulas below are evaluated at every row of Table 1 by the artifact.

At a high level, the construction forms two systems of homogeneous polynomial values and derivatives from the shortened binary public generator. For each proposed locator value, rows that fix otherwise arbitrary scaling and rows that test that locator turn the two systems into an affine feasibility test $Ax = b$. The structural premises stated in Section 7 connect accepted solutions to the hidden Goppa locators; the present section only counts the resulting matrices.

For multiplicity s, one homogeneous jet copy has $\binom{k_\ell+s-1}{s-1}$ columns. Thus the profile gives

$$M = 42\binom{155}{4} + 877\binom{156}{5} = 633{,}864{,}594{,}132. \tag{3}$$

The space of homogeneous degree-seven equations has

$$N_d = \binom{k_\ell + d - 1}{d} = \binom{157}{7} = 407{,}340{,}975{,}756. \tag{4}$$

The four blocks that fix scaling each have length $L_5 = \binom{155}{4} = 23{,}130{,}030$. The largest block that tests a proposed locator is

$$B = \binom{152}{2} + \binom{153}{3} + \binom{154}{4} + \binom{155}{5} = 721{,}656{,}784. \tag{5}$$

Consequently, the affine feasibility matrix has

$$U = 2M = 1{,}267{,}729{,}188{,}264, \tag{6}$$

$$R = 2N_d + 4(L_5 - 1) + 1 + B = 815{,}496{,}128{,}413. \tag{7}$$

The two value-and-derivative systems are binary: the shortened public generator and all their zero-one incidence coefficients, which record the relevant monomial–derivative containments, lie in the two-element field $\mathbb{F}_2$. Counting every potentially incident entry and ignoring further zeros caused by the parity of

Hasse binomial coefficients and by zero coordinates of the public binary points gives the uniform upper bound

$$z_{\mathrm{bin}}^{\mathrm{ub}} = 2N_d \left(42 \sum_{r=0}^{4} \binom{7}{r} + 877 \sum_{r=0}^{5} \binom{7}{r} \right) = 89{,}124{,}576{,}131{,}509{,}776 \tag{8}$$

on the number of binary entries used by the two systems. The exact public-key-specific count can be smaller; the reported tally makes no assumption about the distribution of public point weights. Only the scaling-fixing and locator-testing pieces require arbitrary coefficients from the base field $\mathbb{F}_{2^{12}}$; their two copies have at most

$$z_{\mathrm{gen}} = 2\big(4(L_5 - 1) + 1 + B\big) = 1{,}628{,}353{,}802 \tag{9}$$

nonzero entries. We charge this worst-case count even for zero and infinity locator trials, whose testing rows have fewer nonzero coefficients. Treating both populations as general field multiplications would therefore discard the main circuit-level distinction.

The *Frobenius map* sends each field element x to x^2. Applying the same power of this map to all hidden locators and polynomial coefficients gives an equivalent key, called a *conjugate branch*. A value's *Frobenius orbit* consists of its successive conjugates. A *parity-check matrix* H defines the public code by the equations $Hx = 0$; the solution space is its *kernel*, written $\ker H$.

Choose three coordinates of multiplicity five as reference hold-outs and prescribe three distinct projective rows. A projective linear transformation, represented by an invertible two-by-two matrix up to nonzero scale, can carry any ordered triple of distinct projective points to those rows. Under Assumption 1, the preliminary three-holdout scan at a fourth coordinate accepts exactly one common Frobenius orbit, whose size divides m. Any representative fixes a globally compatible conjugate. After that fourth row is appended, the assumption guarantees compatible *singleton* labels, meaning exactly one accepted value per position, on a set T of $mt + 1$ positions whose parity-check columns have rank mt; it does not require singleton labels at every position. The preliminary scan uses $q + 1$ feasibility systems and the full scan still tests all $n_\ell(q + 1)$ candidates, including positions outside T. The preliminary systems are smaller, so we conservatively charge all of them at the largest four-holdout size:

$$S = (n_\ell + 1)(q + 1) = 3{,}769{,}240. \tag{10}$$

3 Why One Shortened Chart Suffices

The direct construction scans every position of one shortened coordinate set. Its shortening check verifies that the deleted public columns have full rank $|I|$, so the shortened code has dimension $k - |I|$. The source analysis covers the entire public support with four or five such sets, but the known-point key-completion algorithm needs only $mt + 1$ labels expressed in one common projective coordinate system [8]. The following public reduction establishes the rank and capacity of

each chart in Table 1; the conditions for locator recovery require enough unique, compatible labels.

Lemma 1 (One-chart selection). *Let I be the shortening set, let $J = [n] \setminus I$ be the remaining positions, and let H be a full binary parity-check matrix, so the public code is* $\ker H$*. If the shortened systematic construction has dimension $k - |I|$ and $|J| \geq n - k + 1$, then* $\operatorname{rank} H_J = n - k = mt$*. Hence J contains mt positions whose parity-check columns are linearly independent and one further position. If every position in J has a compatible singleton label, these positions supply the required labeled subset.*

Proof. Extending a shortened word by inserting zeros at I identifies $\ker H_J$ with the public code shortened on I. This kernel has dimension $k - |I|$. The rank–nullity theorem gives

$$\operatorname{rank} H_J = |J| - (k - |I|) = n - k = mt.$$

Gaussian elimination over $\mathbb{F}_2$ publicly selects mt independent columns. The inequality $|J| \geq mt + 1$ leaves at least one further position.

When singleton labels are guaranteed only on a subset, the whole-chart rank does not by itself prove that enough independent columns occur among those positions. This is why Assumption 1 separately requires $|T| = mt + 1$ and $\operatorname{rank} H_T = mt$ for compatible singleton labels. Under these conditions, the same public elimination restricted to the singleton positions supplies the completion input, within the full-chart selection budget below.

Table 2. Capacity of one shortened chart for known-point completion.

Parameter set	n_ℓ	mt Required	$mt+1$ Surplus
mceliece348864	919	768	769	150
mceliece460896	1,449	1,248	1,249	200
mceliece6688128	1,886	1,664	1,665	221
mceliece6960119	1,762	1,547	1,548	214
mceliece8192128	1,886	1,664	1,665	221

Lemma 2 (Projective-coordinate completion). *Suppose the labels recovered on one shortened chart have the form $z_i = \phi(\alpha_i^{2^e})$, where the $\alpha_i \in \mathbb{F}_{2^m}$ are distinct hidden locators, e is one common exponent, and ϕ is one common projective linear transformation. Enumerating the projective values absent from any selected $mt+1$ labels includes a value that converts those labels into an affine transform of the same Frobenius conjugate. On that trial the known-point key-completion algorithm receives a valid equivalent binary Goppa representation.*

Proof. Let $p = \phi(\infty)$. Every hidden locator is finite, so p is absent from every z_i and in particular from the selected labels. Choose a projective transformation ψ_p that sends p to infinity. The composition $\psi_p \circ \phi$ fixes infinity and is therefore affine: for some $a \in \mathbb{F}_{2^m}^{\times}$ and $b \in \mathbb{F}_{2^m}$, the transformed labels are $\beta_i = a\alpha_i^{2^e} + b$.

Write $g^{(2^e)}$ for the polynomial obtained by applying $x \mapsto x^{2^e}$ to the coefficients of the hidden monic degree-t Goppa polynomial g. Because binary codeword coordinates are fixed by this field automorphism, applying it coefficientwise to the Goppa syndrome criterion and then substituting $Y = aX + b$ gives

$$\Gamma(\boldsymbol{\alpha}, g) = \Gamma(\boldsymbol{\alpha}^{2^e}, g^{(2^e)}) = \Gamma(\boldsymbol{\beta}, g_p), \qquad g_p(Y) = a^t g^{(2^e)}\left(\frac{Y-b}{a}\right).$$

Thus choosing p as the point at infinity, or *pole*, produces indexed affine labels for an equivalent degree-t binary Goppa key. The deterministic completion theorem of Kirshanova and May applies after Lemma 1 selects more than mt labels whose public parity-check columns have rank mt [8, Theorem 4]. Full public-key verification accepts a valid equivalent key and rejects an invalid output.

To apply the affine recovery algorithm, one projective value must be designated as the pole. For each projective value absent from the selected labels, the completion step chooses it as the pole, converts the other $mt + 1$ labels to affine field values, applies the degree-t Goppa-polynomial recovery algorithm, extends the candidate support, and accepts only a complete public-key verification. By Lemma 2, one trial necessarily chooses a compatible pole. We charge every possible pole in the deliberately loose finite estimate of Section 5; this completion work is smaller than the locator scan in every row. Thus one chart uses $(n_\ell + 1)(q + 1)$ largest-system equivalents, with no multiplier for processing several shortened coordinate sets.

4 A Reliable Affine Sparse Solve

Write $\mathbb{F}_{2^m}$ for the field used by the public construction. Each candidate asks whether the affine linear system

$$Ax = b, \qquad A \in \mathbb{F}_{2^m}^{R \times U}, \quad b \in \mathbb{F}_{2^m}^{R}, \tag{11}$$

is consistent, meaning that some x satisfies every equation. The matrix is sparse because almost all of its entries are zero. A heuristic solver is inadequate for a headline tally: millions of trials amplify even a small probability of failing on a solvable system. We instead use Eberly's reliable scalar Lanczos construction [4]. Scalar Lanczos is an iterative linear-system algorithm that accesses a matrix through products with vectors. Eberly first multiplies rows and columns by random nonzero diagonal entries; this *preconditioning* makes the required rank and solver properties fail only with a bounded probability.

Choose a positive multiple δ of m and work over $\mathbb{F}_{2^\delta}$. Its unique subfield of size 2^m is identified with $\mathbb{F}_{2^m}$. Sample independent diagonal matrices $D_\alpha \in \mathbb{F}_{2^\delta}^{U \times U}$

and $D_\beta \in \mathbb{F}_{2^\delta}^{R\times R}$ with uniformly random nonzero diagonal entries, and sample a uniform vector $z \in \mathbb{F}_{2^\delta}^{U}$. Eberly states the diagonal preconditioner with D_β rather than D_β^2; squaring is a permutation of $\mathbb{F}_{2^\delta}^{\times}$ in characteristic two, so the squared diagonal has exactly the same distribution. His Theorem 4 is stated for blocks of right-hand sides and is used here with block size one, which is odd and not divisible by the characteristic as required. Define

$$G = D_\alpha A^T D_\beta^2 A D_\alpha \tag{12}$$

and apply scalar Lanczos to

$$Gy = D_\alpha A^T D_\beta^2 (b - Az). \tag{13}$$

Return $x = z + D_\alpha y$. A *retraction* is an $\mathbb{F}_{2^m}$-linear map from $\mathbb{F}_{2^\delta}$ back to $\mathbb{F}_{2^m}$ that fixes every element of $\mathbb{F}_{2^m}$; apply it to every coordinate of x. Finally, perform *exact replay*: substitute the retracted vector into the original equation (11) and refuse it unless equality holds.

Theorem 1 (Reliable affine reduction). *If* (11) *is consistent, the right-hand side of* (13) *is the image under G of a uniformly random vector, which is the input distribution required by Eberly's scalar-Lanczos theorem. When the random diagonal scaling preserves the required rank and Lanczos succeeds, the returned vector solves* (11)*. An inconsistent or corrupted output cannot be accepted by exact replay. Moreover, a solution over $\mathbb{F}_{2^\delta}$ can be mapped to a solution over $\mathbb{F}_{2^m}$ by a fixed $\mathbb{F}_{2^m}$-linear retraction.*

Proof. Fix one solution x_0 of (11). Its randomized right-hand side is

$$D_\alpha A^T D_\beta^2 (b - Az) = G D_\alpha^{-1}(x_0 - z).$$

Since z is uniform, the hidden preimage is uniform in $\mathbb{F}_{2^\delta}^{U}$. This is exactly the image-of-a-random-vector distribution in Eberly's theorem. When $\operatorname{rank} G = \operatorname{rank} A$, the kernels of G and AD_α agree. Hence any solution y of (13) satisfies $A(z + D_\alpha y) = b$.

Because $m \mid \delta$, $\mathbb{F}_{2^\delta}$ is an extension of $\mathbb{F}_{2^m}$. Choose an $\mathbb{F}_{2^m}$-linear map $\pi : \mathbb{F}_{2^\delta} \to \mathbb{F}_{2^m}$ with $\pi(1) = 1$, and apply it coordinatewise. It fixes $\mathbb{F}_{2^m}$, so $A\pi(x) = \pi(Ax) = \pi(b) = b$. Finally, exact replay rejects every vector that does not satisfy the original system, regardless of how it arose.

For a matrix with at most U columns and rank at most R, Eberly's bounds give, per system,

$$p_{\mathrm{pre}} \le \frac{11U^2 - U}{2(2^\delta - 1)}, \qquad p_{\mathrm{L}} \le \frac{R(R+1)}{2^\delta}. \tag{14}$$

Here p_{pre} bounds failure of the random diagonal scaling and p_{L} bounds failure of scalar Lanczos. Adding these probabilities for every one-chart system gives a valid upper bound even when the failure events are not independent. For each

parameter set, the artifact selects the least-work multiple of m satisfying $S(p_{\mathrm{pre}}+p_{\mathrm{L}})<2^{-128}$. It compares the rational upper bound with 2^{-128} exactly and checks that the preceding field degree fails this threshold. Integer comparisons also certify the outward-rounded failure exponents printed in the tables; floating-point logarithms do not decide field admission. For the running example this gives

$$\delta = 240, \qquad \log_2\bigl(S(p_{\mathrm{pre}}+p_{\mathrm{L}})\bigr) < -135.1793. \tag{15}$$

This is a bound on algorithmic solver failure. It does not measure the truth of the structural assumptions in Section 7.

5 Bit-Operation Tally

Represent each element of $\mathbb{F}_{2^\delta}$ by the δ binary coefficients of a polynomial. Our direct, or schoolbook, multiplication estimate charges δ^2 Boolean AND operations, $\delta^2-(2\delta-1)$ exclusive-or (XOR) operations to combine equal-degree products, and $\delta(\delta-1)$ XOR operations to reduce the result modulo the fixed irreducible polynomial that defines the field. We therefore charge $C_\times$ for one multiplication and C_{ma} for one multiplication followed by an addition:

$$C_\times(\delta) = 3\delta^2-3\delta+1, \qquad C_{\mathrm{ma}}(\delta) = 3\delta^2-2\delta+1. \tag{16}$$

At $\delta = 240$, these are 172,081 and 172,321 bit operations. This transparent circuit estimate does not use an assumed multiplier for an unspecified field implementation.

One multiplication of a vector by the preconditioned matrix G in (12) is charged by

$$C_G = 2z_{\mathrm{bin}}^{\mathrm{ub}}\delta + (2z_{\mathrm{gen}}+2U+R)C_{\mathrm{ma}}(\delta) = 43{,}357{,}797{,}573{,}481{,}659{,}425. \tag{17}$$

The first term is an upper bound consisting only of XOR operations on bit vectors. The second conservatively charges the arbitrary field coefficients and every diagonal scaling as a full multiply-add. Between matrix products, scalar Lanczos updates a fixed collection of vectors and inner products; we estimate one such recurrence step and the combined initialization and finalization as

$$C_{\mathrm{rec}} = (17U+4\delta+3)C_{\mathrm{ma}}(\delta), \tag{18}$$

$$C_{\mathrm{init}} = (20U+4\delta+3)C_{\mathrm{ma}}(\delta). \tag{19}$$

The coefficients in Eqs. (18) and (19) are retained budget allowances for vector operations, scalar arithmetic, initialization, and finalization. Eberly's theorem supplies the reliability statement, not these operation counts. A detailed implementation trace establishing the sufficiency of these allowances remains to be supplied; the artifact checks their arithmetic rather than proving such a trace. Section B lists each charged contribution separately.

The estimated cost of substituting one candidate into the original base-field equations is

$$C_{\mathrm{replay}} = z_{\mathrm{bin}}^{\mathrm{ub}}m + z_{\mathrm{gen}}C_{\mathrm{ma}}(m) + Rm. \tag{20}$$

The estimated cost of generating the random diagonal entries and affine shift, then applying a general m-by-δ binary linear map to every output coordinate, is

$$C_{\text{source}} = (2U + R + mU)\delta. \tag{21}$$

These costs are included in each feasibility-system row of Table 3. Charging $R+3$ multiplications by G and R recurrence steps gives

$$C_{\text{sys}} = (R+3)C_G + RC_{\text{rec}} + C_{\text{init}} + C_{\text{replay}} + C_{\text{source}} \approx 2^{104.92}. \tag{22}$$

The public one-chart selection estimate for Gaussian elimination is

$$C_{\text{select}} = n_\ell(n-k)^2 + n_\ell(n-k) \approx 2^{29.02}. \tag{23}$$

Put $r = mt = 768$ for the number of parity-check rows and $L = 769$ for the number of selected labels. Enumerating every possible missing pole and, conservatively, permitting even a field inversion for every algebraic operation gives the finite estimate

$$C_{\text{finish}} = (q+1-L)(qL^6 + qnr^4)\, 2mC_\times(m) \approx 2^{94.45}. \tag{24}$$

The inner factors q deliberately allocate one complete sweep through the base field during polynomial splitting and another during support transport. These factors and the powers of L and r are budget envelopes intended to be conservative. The completion theorem establishes polynomial-time recovery under its input conditions; it does not supply the constants in Eq. (24), and an operation-by-operation derivation of that budget remains to be supplied. The key-completion allowance is well below the locator scan and changes no reported headline digit. Summing the charges for all listed stages gives the complete conditional arithmetic estimate

$$C_{\text{total}} = SC_{\text{sys}} + C_{\text{select}} + C_{\text{finish}} \approx 2^{126.77} \text{ bit operations}. \tag{25}$$

Table 3. Detailed conditional attack tally for mceliece348864.

Item	Bit-operation estimate
One affine feasibility system	$2^{104.92}$
All 3,769,240 systems	$2^{126.77}$
Public label selection	$2^{29.02}$
Projective completion and verification	$2^{94.45}$
Complete conditional tally	$\mathbf{2^{126.77}}$

Applying the same formulas to every fixed cell gives Table 4. Work is measured in bit operations and state in simultaneously stored bits. The solver-failure

Table 4. Conditional direct-route arithmetic tally for every Classic McEliece parameter set, with NIST reference levels from [10].

Parameter set	Reference	Solver	Work	State	Failure	Margin
mceliece348864	1 / 2^{143}	$\mathbb{F}_{2^{240}}$	$2^{126.77}$	$2^{51.33}$	$< 2^{-135.17}$	16.23
mceliece460896	3 / 2^{207}	$\mathbb{F}_{2^{260}}$	$2^{145.22}$	$2^{59.10}$	$< 2^{-138.19}$	61.78
mceliece6688128	5 / 2^{272}	$\mathbb{F}_{2^{247}}$	$2^{137.48}$	$2^{55.18}$	$< 2^{-132.51}$	134.52
mceliece6960119	5 / 2^{272}	$\mathbb{F}_{2^{247}}$	$2^{136.65}$	$2^{54.86}$	$< 2^{-133.25}$	135.35
mceliece8192128	5 / 2^{272}	$\mathbb{F}_{2^{247}}$	$2^{137.48}$	$2^{55.18}$	$< 2^{-132.51}$	134.52

column is the union bound over every charged affine system; the margin subtracts the work exponent from the NIST classical-gate reference exponent for the claimed category [10].

Table 5 records the fraction of the modeled total attributable to binary matrix passes, recurrence arithmetic, and the completion allowance. Binary passes account for 90.88% to 96.45% of the total, while completion contributes less than 1.9×10^{-10} of the total in every row. The remaining work mainly consists of general-coefficient matrix products and diagonal scaling. These shares describe the present upper-bound accounting; they do not establish a smaller nonzero count or a reduced attack exponent.

Table 5. Selected contributions to the modeled total; completion is a fraction, not a percentage.

Parameter set	Binary passes	Recurrence	Completion fraction
mceliece348864	90.88%	7.89%	1.87×10^{-10}
mceliece460896	96.45%	3.08%	5.08×10^{-14}
mceliece6688128	95.29%	4.08%	5.74×10^{-11}
mceliece6960119	94.98%	4.35%	6.71×10^{-11}
mceliece8192128	95.29%	4.08%	5.74×10^{-11}

The *active state* is the information that must be stored simultaneously. The model allocates eight vectors of length U for solver data, including recurrence vectors, preconditioning, the right-hand side, and the random shift; two vectors of length R for row data and workspace; and the z_{gen} arbitrary field coefficients. These vector counts are storage allowances whose peak use has not been verified against a concrete implementation. For the running example,

$$C_{\mathrm{state}} = \big(8U + 2R + z_{\mathrm{gen}}\big)\delta \approx 2^{51.33} \text{ bits.} \tag{26}$$

The binary value-and-derivative matrix is not stored: each matrix–vector multiplication regenerates the locations of its binary entries from the public generator and the multiplicity profile. The tally charges the resulting XOR operations but not integer index and address generation. The state budget omits the small

public generator and indexing data as well as implementation-specific buffers. Memory traffic, communication, checkpointing, and redundant execution are not converted into additional bit operations. Thus the arithmetic and state ledgers do not certify a complete implementation's gate count or peak memory.

6 Comparison with Other Proposed Routes

Weis's single-run route. The optimizations announced by Ghoshal et al. [5, Remark 7.3] are developed by Weis [13], whose first revision, received on 11 September 2026, gives explicit constructions, per-parameter estimates, and a storage discussion. The single-run route replaces the millions of separate affine feasibility solves priced here by one binary kernel computation followed by smaller algebraic steps [13, Sec. 5]. Our reduction to one shortened coordinate set does not remove those repeated solves. Table 6 quotes Weis's estimates in the bit-operation model of Ghoshal et al. (GIJS): column (a) keeps the GIJS conditions and cost formulas, adding Weis's proved point-at-infinity statement (Proposition 2), while column (b) also uses the refinements of [13, Sec. 3].

Table 6. Reported $\log_2$ bit-operation estimates. Weis's entries price the main run, under conditional assumptions and accounting different from the present tally [13, Tables 1 and 4].

Parameter set	This route	Weis (a)	Weis (b)
mceliece348864	126.77	114.2	94.0
mceliece460896	145.22	120.3	99.1
mceliece6688128	137.48	123.9	102.2
mceliece6960119	136.65	123.0	101.4
mceliece8192128	137.48	123.9	102.2

Weis estimates that extra constraint columns and local processing add at most a few tenths of a bit to the quoted run costs [13, Table 4]; our column includes all stages budgeted in Eq. (25). The single-run estimates rely on Assumptions 4 and 5, including the claim that the proposed relation families span the entire kernel, and the dimension counts of Steps 2–4; these remain untested at Classic McEliece size [13, Sec. 10 and Table 6]. Column (b) additionally uses Assumptions 1 and 2.

A *distinguisher* tests whether a public code has the hidden algebraic structure rather than looking random. The advertised roughly 20-bit improvement in this test does not transfer in full to our totals: the present cells already account for the reduction by t in the dimension of polynomial products and degree seven for four parameter sets. Finer nonzero counting could also reduce our dominant binary-pass term. At degree seven and multiplicity six, Weis estimates an average of 12.6 nonzeros per monomial and constraint point, against the bound of 120; this uses balanced binary columns and is not a uniform bound

for every public key [13, Sec. 3]. We retain the public-key-independent incidence bound. Removing redundant derivative conditions would also require preserving the affine normalization and locator constraints. Neither saving is included here; a reduction in the binary-pass term would not be the same reduction in the total. Replacing this tally would require compatible arithmetic, storage, and solver-failure accounting.

Weis's gradient route. Weis's *gradient* route uses first partial derivatives of the relations to recover *subcodes*, meaning linear subspaces of the public code. It has reported column-(b) costs of $2^{100.5}$–$2^{110.0}$ bit operations [13, Table 1]. It uses Assumption 3 in place of the single-run Assumptions 4 and 5; column (b) still adds Assumptions 1 and 2. Its finishing steps were tested at full mceliece348864 and mceliece8192128 size using subcodes computed from the secret key, not a public relation computation [13, Secs. 6.2 and 10].

TII-254 key recovery. The TII-254 solution in [11] combines two complete relation kernels, each constrained to vanish at a different public coordinate, called an *anchor*. Known relations not used for locator recovery are removed with exact checks. The associated linear-algebra theorems apply in arbitrary dimensions under their stated hypotheses, but the experiment does not establish the reduced system sizes, useful locator geometry, or success probabilities at Classic McEliece parameters. It therefore supports an alternative recovery route, not a numerical correction to Table 4; its measured GPU times are not a scaling law for these larger instances.

Tangent-space alternatives. Here the hidden polynomial curve parametrizes public columns by locators, up to scaling; its *tangent space* records first-order variation. Previous revisions priced a route that built a separate relation space after omitting each one of several selected coordinates, recovered derivative information there, guessed locator values, aligned Frobenius branches, and interpolated the hidden curve. Its lowest numerical subtotal was larger than Table 3 and omitted the public relation supplier, wrong-guess rejection, support extension, and branch alignment. Weis's comparison with our work quotes these superseded subtotals, approximately 2^{146}–2^{235} bit operations, rather than the current direct-route tally [13, Sec. 8 and Table 5]. Apon's algebraic-geometric obstruction explains why a few guessed locators alone do not identify the binary-Goppa curve [1]. This historical subtotal does not cost the later tangent-space constructions.

On Vedenev's working draft [12] (6 September 2026) : Conjecture 4 requires certain vectors built from pairs of codewords to span a known linear constraint space. Their binary span is denoted by W; the required equality is called *pair-residue coupling completeness*. The displayed length and row-count conditions do not imply this equality. If the support and Goppa polynomial lie in a proper subfield $\mathbb{F}_{2^{m_0}} \subsetneq \mathbb{F}_{2^m}$ and $K = n - 3t - 1$, then $W \subseteq \mathbb{F}_{2^{m_0}}^n$. The residue containment in an ambient space of dimension K [12, App. B] and base change give

$\dim_{\mathbb{F}_2} W \leq m_0 K$, while the conjectured dimension is $mK - (mt + m - 1)$; the latter is strictly larger when $m \geq 2m_0$, $t \geq 1$, and $n > 6t + 1$. This gives counterexample families satisfying both numerical conditions, including $(m, t, n) = (10, 1, 26 \ldots 31)$, $(10, 2, 32)$, and $(12, 1, 32 \ldots 63)$.

For the first two families take the subfield $\mathbb{F}_{32}$, and for the third take $\mathbb{F}_{64}$. When $t = 1$, use $G(X) = X - \gamma$ and any n subfield elements other than γ; for $(10, 2, 32)$, use a monic irreducible quadratic over $\mathbb{F}_{32}$ and all 32 subfield elements. These square-free polynomials have no support roots, and $n > 6t+1$. The binary parity-check rank is at most $m_0 t$, so $k \geq n - m_0 t$. At the respective smallest lengths, the row-count condition follows from $\binom{21}{2} = 210 > 201$, $\binom{22}{2} = 231 > 221$, and $\binom{26}{2} = 325 > 313$, where each right-hand side is $mK - (mt + m - 1)$; the gaps increase with n in the two degree-one ranges. The companion artifact's `THEORY.md` gives the full argument.

This refutes a uniform reading, not a quantified generic statement. A deterministic instance with $(m, t, n, k, D, K) = (12, 64, 985, 217, 856, 792)$ has the predicted rank 8,725, but one instance supplies no success probability. Thus the route still needs a proof of this spanning property for the instance used or a theorem giving its probability, as well as the required first-order flag recovery (reconstructing the nested value and derivative spaces) and a complete cost. The separate `ve26-conj4-lean` artifact checks the linear-algebraic core; the residue containment and constraint-dimension calculation from the dated draft remain inputs to the formalization. Consequently neither the headline tally nor `mccost.py` changes.

Reconstructing only part of a Wiedemann solution. The Wiedemann method solves sparse linear systems by generating sequences of matrix–vector products. Chou and Niederhagen optimize its use in extended linearization (XL), which multiplies polynomial equations and then treats monomials as linear-system unknowns. They reconstruct only the selected coordinates that encode their candidate solution [2]. The exact-replay verification schedule priced here uses the complete solution vector. This is a requirement of the present schedule, not a proof that every possible feasibility test requires full reconstruction. Moreover, our tally uses scalar Lanczos rather than the three-phase Wiedemann cost to which their saving applies. Applying that saving would require a separate solver and verification analysis with compatible reliability and storage accounting, so it does not presently change Table 3.

7 What the Estimate Does and Does Not Establish

The following parts of all five tallies in Table 4 are proved, explicitly estimated, or, for randomized solver failure, given finite probability bounds in this paper:

1. the no-slack multiplicity identity, $d > s$, the power-of-two condition on $s-1$, four available multiplicity-five hold-outs, the exact fixed dimensions, and the separation of nonzero entries into binary and arbitrary-field entries;

2. the reduction of the canonical rank-two relation branch to one rank test on pairwise polynomial products and explicit t-by-t determinants that test the required interpolation maps;
3. the rank-mt one-chart selection lemma and the $mt+1$ label-capacity check for every parameter set;
4. the randomized conversion of each affine system to the form required by Eberly's reliable solver theorem;
5. the linear map back to the base field, exact substitution into the original system, and the aggregate solver-failure bound;
6. the projective-to-affine key reduction, exhaustive pole estimate, deterministic key-completion preconditions, polynomial-basis circuit costs, and active-state estimates.

Canonical-branch reduction. The ambient *generalized Reed–Solomon* code consists of polynomial evaluations up to a fixed degree, with a nonzero multiplier at each locator; it contains the public code. For a fixed shortened instance, put $D = n_\ell - 2t - 1$, let $\Pi(X) = \prod_j (X - \alpha_j)$ be the hidden support polynomial, let $\Gamma(X)$ be the square-free degree-t Goppa polynomial, and let $V \subseteq \mathbb{F}_{2^m}[X]_{\le D}$ be the span of the row polynomials f_i in this ambient representation. Each binary generator row gives the identity $\Pi f_i' + \Pi' f_i = \Gamma^2 f_i^2$; reduction modulo Γ and linearity put all of V in the resulting differential kernel. Consequently, if ϵ is the parity of the relation degree d, every degree-d product belongs to

$$\mathcal{R}_{dD}^{(\epsilon)} = \left\{ h \in \mathbb{F}_{2^m}[X]_{\le dD} : h' + \epsilon \frac{\Pi'}{\Pi} h = 0 \pmod{\Gamma} \right\}.$$

For every $e \ge 2t-1$, the linear map $h \mapsto h' + \epsilon(\Pi'/\Pi)h \bmod \Gamma$ is onto and hence $\dim \mathcal{R}_e^{(\epsilon)} = e + 1 - t$. All five cells in Table 1 satisfy $D \ge 2t - 1$ and $q > D + 1$. For such a cell, suppose the quadratic product map

$$\mu_2 : \mathrm{Sym}^2(V) \longrightarrow \mathbb{F}_{2^m}[X]_{\le 2D}, \qquad \mu_2(f \odot g) = fg, \qquad \operatorname{rank} \mu_2 = 2D + 1 - t.$$

Then the degree-d products span all of $\mathcal{R}_{dD}^{(\epsilon)}$ for every $d \ge 2$. Indeed, the binary identity gives the inclusion above and surjectivity gives its target dimension. Equality in degree two forces the polynomials in V to have greatest common divisor one and to include a degree-D polynomial. Since $q > D + 1$, avoiding the factors of that polynomial and the degree-below-D subspace gives a coprime degree-D pair in V. The product rule for these differential maps and surjectivity of the same first-order map then lift equality successively from degree d to degree $d + 1$.

Let $r = s - 1$, which is a power of two in every cell used here, and for a designated hold-out τ put $Q_\tau(X) = \prod_{j \ne \tau} (X - \alpha_j)^{s_j}$. On polynomials of degree below t, define the following t-by-t map, which tests whether the interpolation conditions away from τ are independent:

$$\mathcal{D}_\tau : h \longmapsto h' + \left(\frac{Q_\tau'}{Q_\tau} + \epsilon \frac{\Pi'}{\Pi} \right) h \pmod{\Gamma}.$$

Under the no-slack identity Eq. (1), nonsingularity of $\mathcal{D}_\tau$ says exactly that the non-hold-out Hermite jet map is invertible. Multiplication by $(X-\alpha_\tau)^r$ preserves the differential kernel because its derivative is zero. Deleting the top held-out jet therefore gives an invertible map and the complete jet map has one relation, unique up to nonzero scaling. Pulling it back through multiplication by 1 and by X^r gives two common rows V_1, V_2 and normalized relations

$$c_\tau^\star = a_\tau(\alpha_\tau^r V_1 + V_2), \qquad (V_2)|_{B_j} = \alpha_j^r (V_1)|_{B_j},$$

with $a_\tau \neq 0$; each relation has held-out value one and zero higher held-out derivatives, and distinct held-out locators make their stack have rank exactly two. The multivariate Hasse chain rule preserves these identities in the public derivative coordinates. It also preserves exact rank two: the product space contains XT^2, whose derivative is nonzero at every support point, so at least one row polynomial has nonzero derivative there and a top multivariate Hasse coefficient survives. Thus the canonical branch follows from the single quadratic rank and the chosen t-by-t determinants. These conditions use the hidden polynomial representation and are therefore finite-instance proof and falsification criteria, not additional public attack steps. Their success probability under the Classic McEliece key distribution remains unproved, and they do not exclude additional public-kernel relations.

The route remains conditional on:

Assumption 1 (Structural assumptions) *For each cell in Table 1, let $E_\ell \in \mathbb{F}_{2^m}^{N_{d,\ell} \times M_\ell}$ be the public value-and-derivative matrix of the shortened binary Goppa instance before the hold-out normalization and locator test. Let B_j be the coordinates of the derivative block used to test the locator at position j, and let π_j project a kernel vector onto that block. We assume the following exact finite-instance conditions, which refine the canonical-form and rank-one-rigidity conjectures and the rank predictions of Ghoshal et al. [5, Conjectures C.2 and C.3 and Sec. 7.4]:*

1. Conditions for canonical relations. *The hidden row-polynomial space has* $\operatorname{rank} \mu_2 = 2D+1-t$, *and $\mathcal{D}_\tau$ is nonsingular for every designated hold-out in the ordered chart used by the attack. The reduction above then supplies the normalized rank-two branch, its held-out values, and its locator-block identity for every affine system used by the scan.*
2. Conditions for locator recovery. *With three prescribed projective rows, the fourth hold-out accepts exactly one common Frobenius orbit, whose size divides m. After fixing any representative branch b, let $\mathcal{C}_j(b) \subseteq \mathbb{P}^1(\mathbb{F}_{2^m})$ be the complete set of labels accepted by the four-holdout feasibility system at position j. There is a set T of $mt+1$ positions such that every $\mathcal{C}_j(b)$ for $j \in T$ is a singleton, the singleton labels are distinct, and the corresponding public parity-check columns have rank mt. These labels are compatible with the hidden support: for one common exponent e and one common projective linear transformation ϕ, they are $\phi(\alpha_j^{2^e})$. The outcome representing infinity*

is allowed, and the fourth-power label used by the equations determines the projective label uniquely because $x \mapsto x^4$ is a field automorphism.

3. Rank requirements. *The actual full and block-projected kernels satisfy*

$$\operatorname{rank}(E_\ell) < N_{d,\ell}, \qquad \dim \pi_j \ker(E_\ell) < |B_j| \quad \textit{for every surviving position } j.$$

These are the two rank conditions required by the conditional recovery theorem. The source's full-matrix and block-deletion rank predictions are used to select the cells in Table 1; this paper assumes that the displayed cells satisfy the two inequalities, but does not need equality with every predicted nullity.

The three premises have different failure modes. Failure of the conditions for canonical relations can remove the true solution, so the attack may refuse rather than incur a hidden finite cost. If the preliminary test leaves more than one Frobenius orbit, the fourth-row choices need not be equivalent. Trying every accepted preliminary value could add a factor of $q+1$, which is not included in the tally. Failure to obtain enough unique locator values can make the search over remaining choices exponential. Let J be the positions with nonempty $\mathcal{C}_j(b)$, and let $\mathcal{T}$ be the family of $(mt+1)$-element subsets $T \subseteq J$ whose public parity-check columns have rank mt. A direct enumeration fallback can choose the most favorable such set, but its candidate-tuple search space still has size

$$N_{\text{direct}} = \min_{T \in \mathcal{T}} \prod_{j \in T} |\mathcal{C}_j(b)|.$$

If every admissible T contains at least r coordinates with two or more surviving labels, then $N_{\text{direct}} \geq 2^r$, which is exponential when $r = \Omega(n_\ell)$. This is a possible obstruction for enumerating all combinations of candidate values, not a lower bound on all ways of combining candidates across coordinates: candidate values at different coordinates may be linked, allowing a more efficient joint search. Such a replacement would have to carry the shared affine solution space, including the relation coefficients, across coordinates, merge equivalent states, and prove that the number of candidate states remains polynomial; neither pairwise candidate counts nor pairwise feasibility proves this. That joint search and its cost are not included in the present tally. Failure of the local projected-kernel inequality can make a coordinate block uninformative and can feed the same obstruction; failure of the full-matrix inequality removes the source theorem's admission guarantee. The source predicts margins of at least 2^{30} dimensions for the full-matrix inequality and at least 2^{23} dimensions for every projected-block inequality in the displayed cells [5, Sec. 7.4, Eq. (11)]. Those margins concern the rank requirements in item 3; they neither imply nor test the conditions for locator recovery in item 2.

Available calibration evidence. Both generated controls below pass the conditions for canonical relations: their quadratic product ranks are $94 = 2 \cdot 49 + 1 - 5$ and $25 = 2 \cdot 13 + 1 - 2$, and their chosen maps $\mathcal{D}_\tau$ have full ranks five and two. On the $(m, n, t, k) = (6, 60, 5, 30)$ control, called M6, degree six and profile

$3^3 4^1 5^{56}$ give $\sum_j s_j = 293 = 6 \cdot 49 + 1 - 5 + 3$, exactly the no-slack count. The 56 include 30 unit columns (each with a single one and all other entries zero), counted at multiplicity $d - 1 = 5$, and the 26 tested non-unit columns. Surplus conditions therefore do not explain its acceptance of multiple locator values, or *nonselectivity*. The literal three-holdout test on the exact public kernel accepted all 65 projective labels at each of 26 tested coordinates. After fixing one coordinate to its matched canonical value, 1,599 of 1,600 two-coordinate tests admitted normalized solutions of rank two with both tested blocks nonzero: 24 coordinates admitted all 64 remaining labels and one admitted 63. Yet two individually accepted pairs had no solution satisfying all three coordinate tests together, with equation rank 109 and augmented rank 110. The pairwise counts therefore cannot be multiplied. Every tested level-four projected-kernel rank was between 11,078 and 11,832, below block size 27,405; the rank requirements alone did not ensure unique locator values.

On the $(m, n, t, k) = (5, 18, 2, 8)$ control, degree seven and profile $5^{13}6^5$ also gave nonselectivity, but both rank inequalities in item 3 fail: the jet matrix has full row rank and the projected kernel fills every locator block. Unless all homogeneous candidates vanish at the normalization coordinate, the candidate dimension after three holdouts must not exceed the rank of the fourth locator block. Neither control is a displayed Classic McEliece cell, and the M6 control uses $(d, s) = (6, 3)$ rather than $(7, 5)$. Nonselectivity has not been observed on any displayed cell.

These negative results concern the local tests for unique values, not global recovery. On the M6 control, a ratio scan at a fourth coordinate with the canonical relation plane supplied from the secret returned exactly six Frobenius conjugates. A different public method combining the relation spaces at individual coordinates recovered and freshly verified its complete key. That recovery starts with a full-kernel computation absent from the priced route, so it supplies neither a cost for that alternative nor an exponential obstruction to it.

There is also positive evidence for unique locator recovery. Ghoshal et al. report a test using the predicted kernel structure without computing the full kernel at $(m, t, n) = (7, 10, 107)$, while noting that it cannot detect an unpredicted kernel direction, and an optimized end-to-end run by Weis at $(m, t, n) = (6, 6, 64)$ [5, App. C]. Weis now documents the latter toy matrix's exhaustive feasibility test: three holdouts leave exactly the m normalized conjugates, and fixing the fourth leaves one value at each tested column [13, Sec. 6.3]. This tests recovery of a unique value at each coordinate on that matrix. The optimized construction is analyzed under Assumptions 4 and 5 and its intermediate dimension counts. TII-253, with $(m, t, n) = (8, 9, 214)$ and $k_\ell = 46$, yielded eight branches and a verified key using that optimized construction [13, Sec. 7]; it was not an exhaustive test of the full repeated-solve scan priced here. None of these experiments establishes the conditions for locator recovery at Classic McEliece size.

Under Assumption 1, none of those branch multipliers remains: the preliminary choices form one globally correlated Frobenius orbit, the set T has singleton labels in one common coordinate system, at most $q + 1 - (mt + 1)$ missing poles

are tried, and the known-point completion takes deterministic polynomial time. The sparse solver has only the separately bounded probability of missing a consistent system and cannot accept an inconsistent one after exact replay. Thus Assumption 1 contains every presently known potential exponential obstruction in this route; this statement does not prove the assumptions or rule out an undiscovered obstruction in their underlying algebra. No probability is assigned to them. No target-scale implementation of the full locator scan has produced public relations, locators, a Goppa polynomial, or a Classic McEliece key. Subject to that boundary, the arithmetic budgets for the fixed direct route lie below the corresponding NIST classical-gate reference level for all five Classic McEliece parameter sets.

Acknowledgement

The author wishes to acknowledge CSC – IT Center for Science, Finland, for computational resources.

Use of Artificial Intelligence

These results originate from a systematic effort to implement and improve cryptanalytic attacks on Post-Quantum Cryptography with AI help. We used OpenAI (GPT 6 Astra, GPT-5.6 Sol) and Anthropic (Fable 5.1, Opus 5) models to formulate theory, develop proofs, analyze results, and prepare and run experiments.

References

1. Apon, D.: An algebraic-geometry lower bound against the ePrint:2026/1747 McEliece key-recovery attack. Cryptology ePrint Archive, Paper 2026/1810 (2026), https://eprint.iacr.org/2026/1810
2. Chou, T., Niederhagen, R.: A practical optimization for wiedemann XL. Cryptology ePrint Archive, Paper 2026/1787 (2026), https://eprint.iacr.org/2026/1787
3. Classic McEliece Team: Classic mceliece: conservative code-based cryptography: guide for security reviewers (October 2022), https://classic.mceliece.org/mceliece-security-20221023.pdf, document of official Classic McEliece web site.
4. Eberly, W.: Reliable krylov-based algorithms for matrix null space and rank. In: Gutierrez, J. (ed.) Symbolic and Algebraic Computation, International Symposium ISSAC 2004, Santander, Spain, July 4-7, 2004, Proceedings. pp. 127–134. ACM (2004). https://doi.org/10.1145/1005285.1005305
5. Ghoshal, A., Ishai, Y., Jain, A., Sun, N.: Quasipolynomial cryptanalysis of the McEliece cryptosystem (or: PIR meets McEliece). Cryptology ePrint Archive, Paper 2026/1630 (2026), https://eprint.iacr.org/2026/1630
6. Hemmert, T.: Key recovery for the McEliece cryptosystem using higher-order vanishing. Cryptology ePrint Archive, Paper 2026/1339 (2026), https://eprint.iacr.org/2026/1339

7. Hemmert, T., Wiemers, A.: Distinguishing Goppa codes using higher-order vanishing. In: Heninger, N., Rosulek, M. (eds.) Advances in Cryptology – CRYPTO 2026. 46th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 17–20, 2026, Proceedings, Part IV. Lecture Notes in Computer Science, vol. 16803, pp. 191–223. Springer (2026). https://doi.org/10.1007/978-3-032-35398-6_7, https://eprint.iacr.org/2025/1661
8. Kirshanova, E., May, A.: Breaking Goppa-based McEliece with hints. Inf. Comput. **293**, 105045 (2023). https://doi.org/10.1016/J.IC.2023.105045, https://eprint.iacr.org/2022/525
9. McEliece, R.J.: A public-key cryptosystem based on algebraic coding theory. Progress Report DSN PR 42-44, Deep Space Network (1978), https://ipnpr.jpl.nasa.gov/progress_report2/42-44/44N.PDF
10. NIST: Submission requirements and evaluation criteria for the post-quantum cryptography standardization process. NIST Call for PQC Standardization Candidates (December 2016), https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf
11. Saarinen, M.J.O.: Two-anchor holdout/hermite: Solving the TII-254 McEliece key recovery challenge. Cryptology ePrint Archive, Paper 2026/1986 (2026), https://eprint.iacr.org/2026/1986
12. Vedenev, K.: Extending distinguishing to key recovery for subfield subcodes of GRS codes. Cryptology ePrint Archive, Paper 2026/1747 (2026), https://eprint.iacr.org/2026/1747
13. Weis, S.A.: Improving GIJS key recovery for classic McEliece. Cryptology ePrint Archive, Paper 2026/1984 (2026), https://eprint.iacr.org/2026/1984

A Artifact Reproduction

Run

```
python3 mccost.py
```

with Python 3.8 or later. This command recomputes all five current-route reports, checks their exact arithmetic and rational solver-failure thresholds, and prints the work shares in Table 5. Each failure exponent is rounded outward and checked by integer arithmetic before it is printed. The artifact intentionally contains no superseded parameter scans or speculative experiments.

B Accounting Ledger and Evidence Status

The artifact separates each system into the six disjoint charges in the first six rows below, then adds public selection and completion. Their sum is exactly Eq. (25). Binary passes here include the binary part of the charged products by G; the separate replay row includes its own base-field binary work. This convention prevents counting replay twice in Table 5.

Work contribution	Total charge over the chart
Binary matrix passes	$S(R+3)\,2z_{\mathrm{bin}}^{\mathrm{ub}}\delta$
General entries and scaling	$S(R+3)(2z_{\mathrm{gen}}+2U+R)C_{\mathrm{ma}}(\delta)$
Scalar recurrence	SRC_{rec}
Initialization and finalization	SC_{init}
Base-field replay	SC_{replay}
Random source and retraction	SC_{source}
Public label selection	C_{select}
Completion and verification	C_{finish}

The state ledger is likewise a sum of three allocations, with the binary matrix generated as needed.

Storage allocation	Bits
Solver-vector allowance	$8U\delta$
Row-data and workspace allowance	$2R\delta$
Arbitrary field coefficients	$z_{\mathrm{gen}}\delta$

Exact dimensions, the binary incidence bound, the field-operation circuit counts, and the solver-failure inequalities have the derivations given in the main text. The recurrence and initialization coefficients, the completion envelope, and the peak vector counts remain explicit budget estimates. The random-source charge is a bit-budget allowance; the cost of obtaining independent uniform nonzero field elements, including sampling retries, is not separately itemized. Reproducing these ledgers checks the sum of the declared charges, but does not prove that an implementation stays within every allocation. Address generation and the other excluded implementation costs must be accounted for before interpreting the total as a complete gate bound.

C Changelog

2026-09-15. Added Weis's conditional single-run and gradient estimates, their accounting limits, and updated evidence for locator recovery. Simplified terminology and clarified the no-slack identity and that Weis's comparison with our work uses superseded tangent-route subtotals. Also cited the certified two-anchor TII-254 recovery and its limits for Classic McEliece costing. The present tally continues to price the fixed direct locator scan; no parameter choice, bit-operation estimate, or state estimate changed.

2026-09-07. Updated the Vedenev discussion to its 6 September draft and recorded that the displayed numerical conditions do not force the coupling equality: proper-subfield instances give explicit admissible counterexample families. A separate Lean artifact checks the linear-algebraic core. No parameter choice, headline bit-operation estimate, or state estimate changed.

2026-09-04. Updated the Vedenev discussion to the 2 September working draft, corrected the overview to require singleton labels only on the required subset,

and separated chart capacity from those recovery conditions. Added work-share and accounting ledgers, made the remaining budget allowances and excluded implementation costs explicit, and replaced floating-point reliability admission with exact rational comparisons and certified outward rounding in the artifact. No parameter choice, headline bit-operation estimate, or state estimate changed.

2026-09-03. Assumption 1 now separates the proved reduction of the canonical branch to finite rank conditions from the unproved conditions for locator recovery and the public rank requirements. Section 7 also records that almost universal pairwise nonselectivity on one small generated instance included an incompatible three-coordinate tuple, so pairwise counts cannot establish exponential growth. No parameter choice, bit-operation estimate, or state estimate changed.

2026-09-02. Section 7 now records the available positive and negative small-parameter evidence for locator selectivity, and Section 6 records the source's announced but unitemized optimizations. Sections 2–5 clarify the binary-entry upper bound, Eberly preconditioner convention, matrix-free state model, and deliberately loose finisher envelope. No parameter choice, bit-operation estimate, or state estimate changed.

2026-09-01. Updated the discussion of Vedenev's 31 August revision and its enumeration-free, all-position tangent-minors alternative. No parameter choice, bit-operation estimate, or state estimate changed.

2026-08-30. Sections 2–5 now give the one-chart direct locator-recovery route, reliable larger-field solves followed by exact maps to the original fields, and one best current estimate for each of the five parameter sets, compared with the cited NIST classical-gate reference levels. Superseded singleton tables and artifact functions were removed; Section 6 reflects Vedenev's current feasibility boundary, and Section 7 records the revised scope and remaining assumptions. We also updated and expanded Assumption 1 later that day.

2026-08-29. Updated for Vedenev's revision dated 2026-08-28: adopted its rank test for rejecting incorrect guesses and the corresponding sensitivity based on the probability that a random matrix loses rank. Selected parameter choices and attack subtotals were unchanged.

2026-08-27. Updated the Introduction and asymptotic discussion for the then-current revisions by Vedenev and by Ghoshal et al.; added the proof for the rank of one point's derivative-row block, exact padding of rectangular systems, active-state accounting, joint parameter optimization, and normalization by projective linear transformations. The artifact then separately reconstructed the published conditional competing baseline without importing its unitemized optimized claim.

2026-08-24. Initial release, followed by same-day cleanup and clarification.