Conditional Subexponential Attacks on Structured Learning with Errors

Tensor Encodings, Noise Distributions, and Comparisons with Lattice Reduction

ChatGPT (with a lot of prompting from Nigel Paul Smart)

Abstract

For exposition we first derive the attack for coefficient-form Ring-LWE in $\mathbb{Z}[X]/(X^n+1)$ with n a power of two and $q=2^t$; later sections extend the method to other structured LWE families under explicit hypotheses. For binary secrets and independent uniform errors in a known interval of W integers, exact reduction to a divisor $D=2^r$ preserves the instance. An explicit Boolean tensor network uses $O(mnr)$ edges for m ring samples. Assuming the tensor-contraction theorem stated in the supplied ETH and ECDLP preprints, it yields time $2^{O(mnr/\sqrt{\log(mnr)})}$ and polynomial space. The algebraic recovery bound is proved without lattice heuristics: the probability of a competing binary secret is at most $[2(\widetilde{W}/D)^m]^n$, where $\widetilde{W}=W$ for even W and $\widetilde{W}=W+1/W$ for odd W. An exact first moment improves the one-sample leading bound to $(2W/D)^n(1+o(1))$; the general multi-sample bound keeps its stated correction. In particular, $q\geq 4W$, $m=1$, and $r=\lceil \log_2 W\rceil+2$ give failure at most 2^{-n}. The derived time bound is subexponential in n when $\log W=o(\sqrt{\log n})$.

The attack bypasses BKZ and is compared with conventional attacks on the original modulus q. We explain BKZ and sieving, derive linear block sizes within an explicit fixed-modulus primal GSA model, and distinguish this calculation from a lower bound on actual BKZ. Astronomical solver cutoffs preclude a concrete crossover. A sharper binary collision bound supports entropy and norm tests for Gaussian-like noise. Extensions cover nonbinary secrets, other moduli and rings, dense LWE, and general lattice reduction.

Keywords: Structured LWE, Ring-LWE, binary secrets, Boolean tensor networks, noise distributions, BKZ, lattice sieving, conditional algorithms

This is an expository derivation for mathematical review, not a claim of a practical attack or a peer-reviewed cryptanalytic result.

Date of this document: 21 November 2026.

Contents

1 Scope, assumptions, and the attack model **4**
1.1 What is and is not being claimed . . . 4
1.2 Notation and precise distribution . . . 4
1.3 The assumed complexity-theoretic interface . . . 5
1.4 Statements proved in the following sections . . . 6

2 The finite chain ring behind the recovery argument **6**

3 Recovery from bounded uniform noise **8**
3.1 A bound valid for every bounded error vector . . . 8
3.2 The slightly sharper uniform-noise bound . . . 8
3.3 Exact first moments and the one-sample refinement . . . 9
3.4 The exact algorithm and its failure convention . . . 11
3.5 Information-theoretic sample requirements . . . 11

4 A full linear-size encoding of negacyclic multiplication **12**
4.1 An auxiliary ring with principal odd-order roots . . . 12
4.2 Packing and the two no-wraparound conditions . . . 13
4.3 Why there are two transform stages, not $\log n$ stages . . . 13
4.4 The actual constraint network . . . 13
4.5 Boolean edge count, degree, descriptions, and space . . . 14
4.6 A smaller bipartite encoding for binary inputs . . . 15

5 Overall conditional complexity and parameter thresholds **17**
5.1 Subexponential in which parameter? . . . 18
5.2 Joint choice of divisor and sample count . . . 19
5.3 The 2^{64} example and what it does not estimate . . . 19
5.4 Verification, success probability, and alternative attacks . . . 19

6 Where the attack sits relative to BKZ and sieving **20**
6.1 Lattices, projections, and the SVP subproblem . . . 20
6.2 Why sieving appears inside the cost estimate . . . 20
6.3 The standard primal lattice embedding . . . 21
6.4 The common heuristic quality model, with its assumptions exposed . . . 21
6.5 Overall cost accounting and the correct comparison . . . 22
6.6 Original versus reduced modulus: which instance is the benchmark? . . . 23
6.7 Fixed modulus and noise: what the primal model actually predicts . . . 23
6.8 Why a block cannot simply be called a smaller Ring-LWE instance . . . 25

7 The cutoff and the crossover against a BKZ model **26**
7.1 An asymptotic rate is not a numerical running-time estimate . . . 26
7.2 Eventual advantage: separate the fixed family from varying parameters . . . 27
7.3 What the supplied source actually says about starting sizes . . . 27
7.4 Concrete sizes and honest conclusions . . . 28

8 Generalization 1: moduli that are not powers of two **29**
8.1 A sufficiently large two-part already suffices . . . 29
8.2 An odd prime divisor: a different image argument . . . 29
8.3 Combining several small odd factors . . . 30
8.4 Where the non-power-of-two case remains difficult . . . 31

9 Generalization 2: other defining polynomials 31
9.1 Compact multiplication extends to every public monic polynomial 31
9.2 A positive case: irreducibility modulo two 32
9.3 A broader sufficient condition for squarefree reductions 32
9.4 A concrete obstruction to the original chain-ring proof 33

10 Generalization 3: secrets that are not binary 33
10.1 Bounded alphabets 33
10.2 Unrestricted secrets can instead be eliminated 33

11 Generalizations 4 and 5: dense LWE and lattice reduction 35
11.1 Why the direct dense-LWE encoding is too large 35
11.2 Sparse public matrices are a genuine special case 35
11.3 Why this is not a generic SVP algorithm 36
11.4 Approximation, reduction direction, and remaining work 36

12 Uniform versus Gaussian-like noise 37
12.1 Same support, different probabilities 37
12.2 Why discarding each coefficient's tails can fail 37
12.3 A sharper binary-secret collision bound for any acceptance set 38
12.4 Entropy gives an effective width and a near-matching threshold 39
12.5 A compact norm test and its full reliability bound 40
12.6 The norm test preserves the tensor size bound 41
12.7 How much concentration helps at the same hard bound 42
12.8 Approximate samplers, asymptotic families, and BKZ 43
12.9 Residue support can matter independently of Gaussian shape 43

13 Conclusions and a checkable interpretation 43

A Verification roadmap and reproducible checks 45
A.1 Interfaces and invariants to audit 45
A.2 Reference pseudocode for the algebraic verifier 46
A.3 A small explicit chain-ring example 46
A.4 Checks performed during preparation 46
A.5 A useful direct counterexample to an invalid shortcut 47

B Source assumptions, versions, and notation map 47

References 48

1 Scope, assumptions, and the attack model

1.1 What is and is not being claimed

The intended reader knows the LWE problem and polynomial-ring arithmetic, but need not know how the cost of BKZ is assembled from smaller shortest-vector computations. The purpose is to supply a checkable reduction, including the bit-level encoding that makes the complexity claim meaningful.

The initial focus on binary-secret Ring-LWE in $\mathbb{Z}[X]/(X^n+1)$, with both n and the coefficient modulus powers of two, is chosen purely for expository organization. This case gives a particularly transparent route through the three ingredients: a collision bound, a compact arithmetic verifier, and the assumed tensor solver. It is the starting point of the presentation, rather than the full scope of the method. Sections 8 and 9 treat other moduli and rings; Sections 10 and 12 broaden the secret and noise distributions; Section 11 includes a sparse-matrix LWE case. Each extension states its own hypotheses and recovery or distinguishing guarantee. The power-of-two hypotheses are used in the initial proofs, but are not asserted to be necessary for every application of the method.

We distinguish three logically different assertions throughout:

1. Algebraic and probabilistic assertions about the input distribution. These are unconditional and proved in this manuscript.
2. Runtime assertions conditional on a specified tensor-contraction algorithm. We assume that algorithm exactly as requested.
3. Comparisons with BKZ and sieving. A cost identity is exact, but its usual numerical instantiation uses explicit lattice heuristics. Those heuristics are not consequences of refuting ETH.

The initial result is a search algorithm for the stated Ring-LWE family; the later extensions establish additional structured cases. These results do not provide a subexponential algorithm for arbitrary LWE, arbitrary SVP, or arbitrary BKZ blocks. In particular, substituting the new runtime for the SVP cost inside every BKZ call would be unjustified. Section 6 explains this distinction before any crossover calculation is made.

1.2 Notation and precise distribution

All logarithms in our analysis are base two; ln denotes the natural logarithm. For the initial expository model, write $n = 2^k$ with $k \geq 1$, $q = 2^t$ with $t \geq 1$, and

$$R = \mathbb{Z}[X]/(X^n+1), \qquad R_Q = R/QR.$$

We identify R_Q with coefficient vectors in $(\mathbb{Z}/Q\mathbb{Z})^n$ in the power basis $1, X, \ldots, X^{n-1}$. Every statement about bounded or binary coefficients refers to this basis. It is not an invariant assertion about an arbitrary basis of the associated lattice. The uninteresting case $n = 1$ can be handled as a scalar problem and is excluded from the ring lemmas.

Let $I = \{\ell, \ell+1, \ldots, \ell+W-1\}$ be a known interval, where $2 \leq W < q$. A secret $s \in \{0,1\}^n \subset R$ is chosen independently of the public multipliers and errors. The recovery theorem holds for every fixed binary s; uniformity of s is needed only for the information-theoretic converse. For each $j \in \{1, \ldots, m\}$ sample

$$a_j \leftarrow R_q \text{ uniformly}, \quad e_j \leftarrow I^n \text{ uniformly}, \quad b_j = a_j s + e_j \pmod{q}. \tag{1.1}$$

All multipliers and all error coefficients in this model are independent. One *ring sample* consists of two degree-below-n polynomials; it contains n scalar equations, but the corresponding scalar rows are correlated rotations, not independent dense LWE rows.

The interval location causes no difficulty. With $U_n = 1 + X + \cdots + X^{n-1}$, replace b_j by $b_j - \ell U_n$; then the errors lie in $\{0, \ldots, W-1\}^n$. Alternatively retain the original interval in every membership predicate. Our bounds are independent of its location and allow its residues to wrap across zero. The case $W = 1$ has known deterministic error and can be treated by ordinary linear algebra, so the noisy statements use $W \geq 2$.

For ciphertexts with an additional message term, that term must be known, removed, or included in the constraints. If it is a multiple of the reduced modulus, it disappears under the exact projection used below. We do not assume an arbitrary unknown plaintext is small noise, nor that evaluated ciphertexts have the fresh independent distribution (1.1).

1.3 The assumed complexity-theoretic interface

A Boolean tensor network is a finite undirected multigraph whose edges carry bits. A vertex is a Boolean predicate of its incident bits. Its contraction is the number of assignments satisfying all predicates. Parallel edges represent separate ports. The degree is the number of incident Boolean ports, not the number of ring-element words.

Assumption 1.1 (Tensor counting interface)**.** For networks with s Boolean edges, polynomially many vertices, polynomial total description length, degree $O(s^{9/10})$, and uniformly polynomial-time evaluable predicates, exact contraction is computable in time $2^{O(s/\sqrt{\log s})}$ and polynomial space.

This is the interface extracted in Theorem 2 of the ECDLP manuscript [3], from Theorem D.1 of [1]. The latter permits a polynomial description factor and more general local arithmetic; polynomial Boolean predicates are a permitted special case. A generic network can meet its bipartite requirement by subdividing each edge with an equality predicate, doubling the edges while preserving the count and degree bound. The layered constructions in Section 4 are already bipartite, so this generic normalization is unnecessary for them. To obtain an exact integer count, interpret Boolean predicates modulo 2^P, with P one greater than the number of edges before subdivision: the count is at most 2^{P-1}, so no information is lost modulo 2^P.

For precision, if E is our actual edge count and Δ its degree, the original theorem's size parameter may be set to

$$s_\star = \max\{E_{\mathrm{bip}}, \left\lceil \Delta^{10/9} \right\rceil, 2\}, \qquad E_{\mathrm{bip}} = \begin{cases} E, & \text{already bipartite,} \\ 2E, & \text{otherwise.} \end{cases}$$

This accommodates any needed subdivision and the exact degree cap. Precision $P = E + 1 \leq 2s_\star$ is linear in the source size parameter. In our regime $s_\star = O(mnr)$; it is not silently identified with the number of secret bits. Polynomial preprocessing and input reading are added separately.

The supplied ETH refutations are not used as a slogan that every NP problem becomes easy. The fixed-width SAT bounds in [1, 2] are measured in the number of SAT variables or clauses. A polynomial reduction that changes n to n^2 need not preserve subexponentiality in n. Our proof therefore constructs a nearly linear-size tensor encoding explicitly. We use the direct tensor interface rather than infer it from a generic circuit-to-SAT transformation.

The official formalization scope describes a fixed-width CNF counting bound with exponent $O_k(n(\log\log n)^2/\sqrt{\log n})$ [14]. That statement is distinct from the succinct tensor interface used here. We do not identify the latter with a Lean-verified theorem; Assumption 1.1 remains the explicit dependency.

Lemma 1.2 (Counting to search)**.** *Suppose a network represents candidate secrets, each with exactly one satisfying extension to auxiliary edges. Then one can count candidates, detect uniqueness, and recover a unique candidate with at most $n + 1$ counting calls and polynomial overhead.*

Proof. The first count is the number of candidate secrets by the unique-extension property. If it is not one, report fail. Otherwise fix secret bits in order. To fix a bit to zero, add that condition to one predicate incident to the edge carrying the bit; no edge or port is added. Count again. If the count is positive keep zero, and otherwise set the bit to one. A unique completion remains at every step. After n such queries every secret bit is fixed. Counts need at most $O(E)$ bits. Reusing the same graph with different predicate constants keeps the same size and degree. □

1.4 Statements proved in the following sections

The recovery proof has three components: a chain-ring lemma, a collision bound for bounded secrets, and a deterministic exact search procedure. The encoding proof is separate: a triadic auxiliary-ring transform checks negacyclic multiplication with $O(nr)$ edges and degree $O(n^{3/4}r)$. Only after both proofs do we apply Assumption 1.1. No Gaussian heuristic, GSA, random-lattice assumption, lattice sieve, or BKZ success heuristic enters those arguments.

Our source assumptions do not furnish usable numerical constants for cryptographic dimensions. In particular, their asymptotic cutoff is not a concrete security estimate. Section 7 makes the crossover question explicit rather than replacing an unknown constant by one.

2 The finite chain ring behind the recovery argument

Fix $D = 2^r$ and let $T = R_D$. The following proof establishes the required ideal structure directly; no theorem about algebraic number fields is needed.

Lemma 2.1 (Local structure and ideals)**.** *Let $n = 2^k \geq 2$. In T, put $Y = X + 1$. Then:*

1. *An element is a unit precisely when its image under $T \to T/(2, Y) \cong \mathbb{F}_2$ is nonzero.*
2. *$2 = uY^n$ for some unit $u \in T$.*
3. *The ideals of T are exactly (Y^j), for $0 \leq j \leq nr$; $Y^{nr} = 0$ and $|(Y^j)| = 2^{nr-j}$.*
4. *For $h \leq r$, $(2^h) = (Y^{nh})$. In particular, (2) consists exactly of vectors with even coefficients in the X basis.*

Proof. In characteristic two, repeated squaring gives $(X + 1)^n = X^n + 1$. Consequently

$$T/2T \cong \mathbb{F}_2[Y]/(Y^n).$$

An element of the latter ring with constant term one is invertible: write it as $1 + z$ with $z \in (Y)$ and use the finite geometric series for its inverse. If $fv = 1 \pmod 2$ in T, write $fv = 1 + 2w$. Since $(2w)^r = 0$, the finite series $\sum_{j=0}^{r-1}(-2w)^j$ inverts $1 + 2w$. Thus every element nonzero modulo $(2, Y)$ is a unit. Elements of $(2, Y)$ cannot be units because their image in the nonzero quotient field is zero. This proves the first assertion and also proves that $(2, Y)$ is the unique maximal ideal.

Since n is even, expanding $(Y - 1)^n + 1$ gives

$$(Y - 1)^n + 1 = Y^n + 2h(Y), \qquad h(0) = 1.$$

All intermediate binomial coefficients are even, as also follows from the characteristic-two identity above. The defining relation in T is $Y^n = -2h(Y)$. The first assertion makes h a unit, proving $2 = uY^n$. It follows that $(2, Y) = (Y)$ and $Y^{nr} = 0$.

For a nonzero element f, choose the largest v for which $f \in (Y^v)$; such a $v < nr$ exists. Write $f = Y^v g$. If g were not a unit, it would lie in (Y), contradicting maximality of v.

Hence $(f) = (Y^v)$. For any nonzero ideal choose an element with smallest such v. Every element of the ideal lies in (Y^v), while the chosen element generates (Y^v), so the ideal equals (Y^v). The zero ideal is (Y^{nr}).

Multiplication by Y^j surjects $T/(Y) \cong \mathbb{F}_2$ onto $(Y^j)/(Y^{j+1})$, so each successive quotient has at most two elements. There are nr such quotients and $|T| = 2^{nr}$, because the defining polynomial is monic of degree n. All successive quotients must therefore have exactly two elements. This also proves that no earlier power is zero and gives the stated cardinalities. Finally 2^h differs from Y^{nh} by a unit, and coefficientwise reduction modulo two proves the last claim. □

Corollary 2.2 (Bounded differences generate a large image)**.** *Let $H = 2^h < D$. If a polynomial δ has at least one coefficient not divisible by H, then*

$$\delta T \supseteq HT, \qquad |\delta T| \geq (D/H)^n.$$

For two distinct binary vectors s, s', this holds with $H = 2$.

Proof. The coefficient basis shows that $\delta \notin HT$. The principal ideal (δ) is therefore not contained in HT. The ideals form a chain, so it contains HT. The latter has $(D/H)^n$ elements. A nonzero binary difference has coefficients in $\{-1, 0, 1\}$ and hence has an odd coefficient. □

This is stronger than merely knowing the rank of multiplication modulo two. That rank can be very small. Nevertheless its lifted image contains every even coefficient vector. The independent higher bits supply the collision bound. Ignoring them would incorrectly suggest that nilpotents defeat the argument.

Lemma 2.3 (Uniform images and parity classes)**.** *For fixed $\delta \in T$, if a is uniform in T, then $a\delta$ is uniform in the additive subgroup δT. If $\delta T \supseteq HT$, conditioning on a coset of HT makes all coefficient coordinates independently uniform over the corresponding residue classes modulo H.*

Proof. Multiplication by δ is a homomorphism of finite additive groups. Every nonempty fiber is a translate of its kernel and has the same size. This proves uniformity on the image. A coset $z + HT$ is the Cartesian product of the n scalar classes $z_i + H\mathbb{Z}/D\mathbb{Z}$, each containing D/H elements. Uniformity on this product is exactly the claimed conditional independence. Uniformity on a subgroup containing HT is a mixture of these uniform cosets. □

Lemma 2.4 (Unit testing and lifting)**.** *An element of R_{2^t} is invertible if and only if the sum of its coefficients is odd. Exactly half of all elements are units. Being a unit is unchanged upon reduction modulo 2^r, for $1 \leq r \leq t$. An inverse, when it exists, is computable in polynomial time in n, t.*

Proof. The map to $T/(2, X+1)$ evaluates X at one modulo two, giving the coefficient sum. Lemma 2.1 proves the criterion, and the surjective linear map to $\mathbb{F}_2$ has two equal-sized fibers. Modulus reduction does not change this parity. An inverse modulo two can be found by the polynomial extended Euclidean algorithm, or by Gaussian elimination on the multiplication matrix. If $uv = 1 \pmod{2^h}$, replacing v by $v(2 - uv)$ squares the error and gives an inverse modulo 2^{2h}. Repeated doubling reaches t with polynomially many bit operations. □

Lemma 2.5 (Exact modulus projection)**.** *For $D \mid q$, reduction $R_q \to R_D$ is a surjective ring homomorphism and maps a uniform element to a uniform element. It maps $b = as + e \pmod{q}$ to the identical equation modulo D, with no rounding term. If a coefficient is known to belong to an interval of length at most D, its residue modulo D has a unique lift to that interval.*

Proof. Integer coefficient reduction respects sums and products and the same polynomial relation. Each residue has $(q/D)^n$ preimages. Two different integers in an interval of length at most D differ in absolute value by less than D and cannot be congruent modulo D. □

This projection is different from rounded modulus switching to an unrelated modulus. In particular, it does not multiply a rounding error by the secret. For a binary secret, reduction is already injective on the secret alphabet as soon as $D \geq 2$. There is consequently no need to reconstruct the full secret through a Chinese remainder procedure.

3 Recovery from bounded uniform noise

The main theorem concerns a random *input*, not randomized choices inside the counting routine. Once the samples are fixed, the conditional search procedure is deterministic.

3.1 A bound valid for every bounded error vector

Lemma 3.1 (Interval test on a multiplication image)**.** *Let $H = 2^h < D$ and $\delta T \supseteq HT$. Let I be an interval of $W \leq D$ distinct residues. For every fixed $e \in I^n$,*

$$\Pr_{a \leftarrow T}[e - a\delta \in I^n] \leq \left(\frac{H \lceil W/H \rceil}{D} \right)^n .$$

Proof. Condition on a coset of HT in the multiplication image. In each scalar residue class modulo H, the translated interval $e_i - I$ has at most $\lceil W/H \rceil$ elements. Lemma 2.3 gives probability at most $\lceil W/H \rceil / (D/H)$ in each coordinate. Multiply the probabilities, then average over the cosets. □

For binary differences $H = 2$, so this gives the convenient upper bound $[2 \lceil W/2 \rceil / D]^n$ for a fixed competing secret and one sample. It does not require uniform errors or independence between their coordinates; independence of the errors from the public multiplier suffices.

3.2 The slightly sharper uniform-noise bound

Define

$$\alpha_W = \begin{cases} W, & W \text{ even}, \\ W + 1/W, & W \text{ odd}. \end{cases} \tag{3.1}$$

Lemma 3.2 (Averaging over uniform errors)**.** *For a nonzero binary difference $\delta = s' - s$ and independent uniform $a \in T$, $e \in I^n$, where $W < D$,*

$$\Pr[e - a\delta \in I^n] \leq (\alpha_W / D)^n .$$

Proof. The image contains $2T$ by Corollary 2.2. Let N_0, N_1 be the number of even and odd residues in I, respectively. Their sum is W, and their difference is zero for even W and has absolute value one for odd W. Conditioning on a parity vector of $a\delta$, its higher bits are independent uniform bits. In one coordinate with fixed parity zero, the probability of $e - z \in I$ is

$$\frac{N_0^2 + N_1^2}{W(D/2)} .$$

This counts ordered pairs of interval elements with the same parity. For parity one the numerator is $2N_0N_1$. If W is even both probabilities are W/D. If W is odd they are $(W+1/W)/D$ and $(W-1/W)/D$. The first is an upper bound in either parity. Coordinates are independent under this conditioning, including the independent errors. Taking the product and averaging over the image's parity vectors proves the claim. □

Theorem 3.3 (Binary-secret identifiability)**.** *Fix a binary secret, $D = 2^r \mid q$, $W < D$, and m samples distributed as in (1.1). The probability that a different binary secret satisfies all reduced equations with errors in I^n is at most*

$$(2^n - 1)(\alpha_W/D)^{mn} \le \left[2(\alpha_W/D)^m\right]^n. \tag{3.2}$$

For any error distribution supported in I^n and independent of independent uniform public multipliers, the same assertion holds with α_W replaced by $2\lceil W/2\rceil$.

Proof. Fix a competing binary secret s' and put $\delta = s' - s$. The residual under s' is $e_j - a_j\delta$. Lemma 3.2 bounds its acceptance probability for one sample. For fixed s, s', different sample events are independent, so their probabilities multiply. There are $2^n - 1$ choices of s'. A union bound proves the first assertion without assuming independence between different candidates. For the bounded-noise version, condition on all errors and use Lemma 3.1; the public multipliers remain independent and uniform. □

Corollary 3.4 (A simple one-sample modulus choice)**.** *For $W \ge 2$, put $r = \lceil \log_2 W \rceil + 2$ and $D = 2^r$. If $q \ge 4W$, then $D \mid q$ and one ring sample has a unique consistent binary secret except with probability at most 2^{-n}. This holds for every independent error law supported in the interval, not just the uniform law.*

Proof. Let $H_W = 2^{\lceil \log_2 W \rceil}$. Since q is a power of two at least $4W$, it is at least $4H_W = D$. Also H_W is even, and $\lceil W/2 \rceil \le H_W/2$. The bounded-noise version of (3.2) has base $4\lceil W/2\rceil/D \le 1/2$. □

3.3 Exact first moments and the one-sample refinement

The preceding bound is convenient and valid for every sample count. For uniform errors we can compute the first moment exactly, retaining the correlations between parity coordinates rather than maximizing over them. Put $\mathcal{V}_j = (Y^j) \subseteq \mathbb{F}_2[Y]/(Y^n)$, with $Y = X + 1$ and coordinates still measured in the X basis. Let wt be binary Hamming weight and set

$$A_j(x) = \sum_{v \in \mathcal{V}_{n-j}} x^{\mathrm{wt}(v)}, \qquad \epsilon_W = \begin{cases} 0, & W \text{ even}, \\ 1/W^2, & W \text{ odd}. \end{cases}$$

In particular $A_0 = 1$. The Euclidean binary dual satisfies $\mathcal{V}_j^{\perp} = \mathcal{V}_{n-j}$. To see this, reversing Y^{n-j} multiplies it by a power of X, a unit modulo $X^n + 1$. Products of a multiple of Y^j and a reversed multiple of Y^{n-j} vanish modulo $Y^n = X^n + 1$, so their constant cyclic coefficient, the coordinate inner product, is zero. The respective dimensions $n - j$ and j prove equality of the dual spaces.

Lemma 3.5 (Exact binary valuation strata)**.** *Fix a binary s. For every competitor $s' \ne s$, the chain-ring valuation $v_Y(s' - s)$ is an integer $j < n$. Exactly 2^{n-j-1} competitors have valuation j, and their multiplication ideal has size $D^n/2^j$.*

Proof. A nonzero binary difference has an odd coefficient and is not in $2R_D = (Y^n)$. Its valuation is below n and remains unchanged modulo two: reduction sends Y^j times a unit to Y^j times a unit in $\mathbb{F}_2[Y]/(Y^n)$. As s' varies, $(s' - s) \bmod 2$ ranges bijectively over binary coefficient vectors. The change to the Y basis is invertible. A vector of valuation j has j zero initial Y coefficients, then one, then $n - j - 1$ arbitrary coefficients. The image size is the chain-ring cardinality from Lemma 2.1. □

Theorem 3.6 (Exact first moment for uniform interval errors)**.** *For the distribution in (1.1), let N_{false} count the incorrect binary secrets consistent with all m reduced samples. Then*

$$\mathbb{E}N_{\mathrm{false}} = \left(\frac{W}{D}\right)^{mn} \sum_{j=0}^{n-1} 2^{n-j-1} A_j(\epsilon_W)^m. \tag{3.3}$$

In particular, this expression upper-bounds ambiguity. For one sample, define

$$S_n(x) = \sum_{j=0}^{n-1} 2^{n-j-1} A_j(x) = \sum_{v \in \mathbb{F}_2^n} x^{\mathrm{wt}(v)} (2^{v_Y(v)} - 1), \qquad v_Y(0) = n. \tag{3.4}$$

Then $\mathbb{E}N_{\mathrm{false}} = (W/D)^n S_n(\epsilon_W)$.

Proof. Fix a competitor of valuation j. Its multiplication image contains $2R_D$ and its parity is uniform on $\mathcal{V}_j$. The computation of Lemma 3.2 gives the conditional coordinate acceptance factors $(W/D)(1 + \epsilon_W(-1)^{p_i})$. Expanding their product and averaging over $p \in \mathcal{V}_j$ leaves exactly those monomials whose indicator vector is in $\mathcal{V}_j^{\perp}$. Indeed a nontrivial character averages to zero by pairing p with a translate on which its sign changes. Thus the exact one-sample probability is $(W/D)^n A_j(\epsilon_W)$. Independence across samples and Lemma 3.5 give (3.3). For the second equality in (3.4), exchange the sums: a nonzero vector of valuation ν belongs to $\mathcal{V}_{n-j}$ exactly for $n - \nu \le j \le n - 1$, whose weights sum to $2^\nu - 1$. The zero vector contributes $2^n - 1$. □

Lemma 3.7 (Recurrence and asymptotics of the valuation enumerator)**.** *For $n = 2N$ and $0 \le x \le 1$,*

$$S_{2N}(x) = 2^N S_N(x^2) + (1 + x^2)^N S_N\left(\frac{2x}{1 + x^2}\right), \qquad S_1(x) = 1. \tag{3.5}$$

Also $S_n(0) = 2^n - 1$, $S_n(1) = (n/2)2^n$, and $R_n(x) := S_n(x)/2^n \to 1$ for every fixed $x < 1$, uniformly on each interval $0 \le x \le x_0 < 1$.

Proof. Write a length-$2N$ vector as (u, w) and put $h = u + w$. If $h = 0$, its polynomial is $(1 + X^N)u = Y^N u$, and its weight is $2\,\mathrm{wt}(u)$. This part of the valuation sum equals $2^N S_N(x^2) + (2^N - 1)(1 + x^2)^N$. If $h \neq 0$, its valuation equals $v_Y(h) < N$. Summing over w for fixed h gives $(2x)^{\mathrm{wt}(h)}(1 + x^2)^{N - \mathrm{wt}(h)}$. The remaining contribution is $(1 + x^2)^N[S_N(2x/(1 + x^2)) - (2^N - 1)]$. Adding proves the recurrence. The values at zero and one follow from (3.4) and $A_j(1) = 2^j$.

All coefficients of S_n are nonnegative. Thus $R_n(x) \le n/2$ on $[0, 1]$, and $R_n(x) \ge 1 - 2^{-n}$. Normalize the recurrence and use $2x/(1 + x^2) \le 1$ to get

$$R_{2N}(x) \le R_N(x^2) + \frac{N}{2}\left(\frac{1 + x^2}{2}\right)^N.$$

Choose a power of two M with $\sqrt{n}/2 < M \le \sqrt{n}$. Iterating to length M bounds the sum of error terms by $(n \log_2 n/4)c_x^M$, where $c_x = (1 + x^2)/2$. Every nonconstant term of S_M has weight at least two, since positive valuation implies even coefficient sum. Hence $R_M(y) \le 1 + (M/2)y^2$. Altogether

$$1 - 2^{-n} \le R_n(x) \le 1 + \frac{M}{2} x^{2n/M} + \frac{n \log_2 n}{4} c_x^M. \tag{3.6}$$

For fixed $x < 1$ both added terms vanish. The same upper bound with x_0 proves uniformity. The finite small values of n do not affect the limit. □

Consequently, for one sample,

$$\Pr[\text{ambiguity}] \le \left(\frac{2W}{D}\right)^n R_n(\epsilon_W) = \left(\frac{2W}{D}\right)^n (1 + o(1)).$$

This removes the odd-W exponential penalty in the one-sample bound. The finite factor R_n must still be retained when certifying a numerical failure probability; it can exceed one. For even W, the exact first moment is simply $(2^n - 1)(W/D)^{mn}$ for every m.

Why this simplification is not a general multi-sample bound. The powers $A_j(\epsilon_W)^m$ in (3.3) cannot be dropped. For example set $n = 64, m = 8, W = 3, D = 8$, take $s = 0$, and consider the all-ones competitor. It has valuation $n - 1$, whose dual code is the even-weight code. Its one-sample acceptance probability is

$$p = \tfrac{1}{2}[(5/12)^{64} + (1/3)^{64}].$$

The ambiguity probability is at least p^8, and exact arithmetic gives $p^8/[2^{64}(3/8)^{512}] > 56$. Thus replacing α_W by W in the general bound (3.2) would be false. Use the exact moment, or retain that convenient coarse bound, for general m.

If $H_W = 2^{\lceil \log W \rceil}$ and $W/H_W \le 1 - \delta$ with fixed $\delta > 0$, and $2H_W \mid q$, the smaller choice $D = 2H_W$ gives one-sample ambiguity at most $(1 - \delta)^n R_n(\epsilon_W)$. It saves one retained bit when that reliability is sufficient. Some such cases already follow from the older α_W bound; Corollary 3.4 remains the simple distribution-independent choice with its stated 2^{-n} guarantee.

3.4 The exact algorithm and its failure convention

Algorithm 1: Binary Ring-LWE recovery.

1. Choose an integer $r \le t$ with $2^r > W$ and reduce all samples exactly modulo $D = 2^r$.
2. Build the network of Section 4 for the conjunction $s \in \{0, 1\}^n$ and $b_j - a_j s \in I^n \pmod D$ for all j.
3. Count its solutions exactly. If the count is not one, return fail.
4. Use Lemma 1.2 to recover the unique binary secret. Interpret its bits as integers, hence also as the original secret modulo q.
5. Optionally compute the original residuals modulo q and check membership in the known interval. This is a polynomial-time consistency check, not an additional heuristic.

The true secret is always feasible under the bounded-noise promise. Thus whenever the algorithm outputs a secret it is the true one; ambiguity causes a reported failure, not an incorrect answer. The probability of failure is bounded by Theorem 3.3. Additional fresh samples can be added if available, increasing the network size and reducing the bound. Repeating the counting computation on the same ambiguous input does not change its ambiguity.

3.5 Information-theoretic sample requirements

Proposition 3.8 (A converse independent of computation)**.** *For uniform binary s, m ring samples, and independent uniform noise of width $W \le q$, every recovery algorithm has success probability at most*

$$\min\{1, [(q/W)^m/2]^n\}. \tag{3.7}$$

This holds even when computation is unlimited and for every fixed choice of the public multipliers.

Proof. Condition on the public multipliers. Each of the $2^n W^{mn}$ pairs consisting of a secret and all errors has the same probability. For a fixed secret and a fixed tuple of labels, there is at most one error tuple in the interval, since $W \le q$. Hence each feasible joint event $(s, b_1, \ldots, b_m)$ has probability $1/(2^n W^{mn})$. For each label tuple, an optimal deterministic decoder can select at most one secret, and there are at most q^{mn} label tuples. Randomization cannot improve the optimum of this linear success functional. Summing gives (3.7), together with the trivial upper bound one. □

For even W, the sufficient uniqueness condition $m \log_2(D/W) > 1$ matches this information threshold up to the desired positive reliability margin. For odd W the correction $1/W$ is small. For example, $m \log_2(D/\alpha_W) \geq 2$ gives failure at most 2^{-n}. If $W = q$, the noise is uniform modulo q, and the labels reveal nothing about the secret. Uniform noise of width a multiple of q has the same problem after reduction. Our interval-support algorithm is stated for $W < D \leq q$ and does not claim recovery from such informationless inputs.

4 A full linear-size encoding of negacyclic multiplication

This section proves the part of the reduction that a generic invocation of an SAT algorithm would not establish. A schoolbook circuit has too many wires, while one predicate reading the whole witness has too many ports. We adapt the one-split Fourier method used for integer multiplication in [3] to coefficients modulo a power of two. A triadic transform is needed because division by two is unavailable. Such auxiliary-ring transforms are classical polynomial-multiplication tools [7, 8]; the Boolean edge accounting is supplied here.

4.1 An auxiliary ring with principal odd-order roots

Let J be the smallest power of three with $J^2 \geq n$, and put

$$B = \lceil n/J \rceil, \quad M = 2J, \quad A = (\mathbb{Z}/2^r\mathbb{Z})[Y]/(Y^{2J} + Y^J + 1).$$

An element of A is represented by M canonical coefficients of r bits each, so its word size is $w_A = Mr = 2Jr$. There are no alternative bit representations of a coefficient: every r-bit string represents exactly one residue modulo 2^r. Choose L to be the smallest power of three at least $2B - 1$. Since $B \leq J$, we have $L \leq 3J$, and therefore L divides $3J$. Set

$$\omega = Y^{3J/L} \in A. \tag{4.1}$$

Lemma 4.1 (Principal root and inversion)**.** *The element ω is a principal Lth root: $\omega^L = 1$, and $\sum_{i=0}^{L-1} \omega^{hi} = 0$ for every $0 < h < L$. Also L is a unit in A. Thus for $x \in A^L$ the transform*

$$\mathrm{DFT}_\omega(x)_t = \sum_{j=0}^{L-1} x_j \omega^{jt}$$

has inverse $L^{-1}\,\mathrm{DFT}_{\omega^{-1}}$.

Proof. The defining relation implies $Y^{3J} = 1$ by multiplication with $Y^J - 1$. This proves $\omega^L = 1$. If $L = 1$ the sum condition is empty. Otherwise write $h = 3^a h_0$ with $3 \nmid h_0$ and $0 \leq a < \log_3 L$. The geometric sum splits into repeated cycles of length $L/3^a$. In each cycle, group exponents separated by one third of that length. The resulting factor is

$$1 + Y^{Jh_0} + Y^{2Jh_0} = 0,$$

because h_0 is congruent to one or two modulo three. This proves the principal-root condition without assuming that nonzero elements are units. The scalar L is odd, so it has an inverse modulo 2^r, hence in A. For inversion, expand the proposed inverse:

$$L^{-1} \sum_{t<L} \sum_{j<L} x_j \omega^{(j-i)t} = x_i,$$

using the principal-root sums. This proves the inverse formula over the ring with its zero divisors. □

4.2 Packing and the two no-wraparound conditions

For degree-below-n polynomials a, u, write

$$a(X) = \sum_{i<B} X^{iJ} a_i(X), \qquad u(X) = \sum_{i<B} X^{iJ} u_i(X),$$

where each block has degree below J, and the last block is padded with zeros. Interpret the blocks as elements of A by replacing X by Y, and pad the block arrays with zeros to length L. Their ordinary block convolution is

$$C_t(Y) = \sum_{i+j=t} a_i(Y) u_j(Y), \qquad 0 \le t \le 2B-2. \tag{4.2}$$

Each summand, and hence C_t, has degree at most $2J-2 < M$. Therefore reduction modulo $Y^{2J} + Y^J + 1$ does not alter the polynomial C_t other than reducing its scalar coefficients modulo 2^r. Also $L \ge 2B-1$ prevents cyclic wraparound in the block index. Evaluating the block polynomials at powers of ω, multiplying pointwise, and applying the inverse transform recovers exactly the $C_t \in A$. Coefficients with $t > 2B-2$ are zero.

There is no need for guard digits of width $\log n$. In integer packing, one must prevent scalar carries from mixing coefficient positions. Here we work in a polynomial ring with independent coefficients modulo 2^r; large sums are supposed to reduce modulo 2^r. The two degree bounds above are the only wraparound issues.

4.3 Why there are two transform stages, not $\log n$ stages

Factor $L = L_1 L_2$, where $L_1 = 3^{\lfloor \log_3 L/2 \rfloor}$ and $L_2 = L/L_1$. Thus $\max(L_1, L_2) \le \sqrt{3L}$. For $j_2 < L_2$ and $t_1 < L_1$ define

$$y_{j_2,t_1} = \omega^{j_2 t_1} \sum_{j_1<L_1} (\omega^{L_2})^{j_1 t_1} x_{L_2 j_1 + j_2}, \tag{4.3}$$

$$z_{t_1+L_1 t_2} = \sum_{j_2<L_2} (\omega^{L_1})^{j_2 t_2} y_{j_2,t_1}. \tag{4.4}$$

Expanding the right side gives the DFT: with $j = L_2 j_1 + j_2$ and $t = t_1 + L_1 t_2$, the exponent jt differs from $L_2 j_1 t_1 + j_2 t_1 + L_1 j_2 t_2$ by $L j_1 t_2$, a multiple of L. This proves the factorization directly.

One vertex for each j_2 checks all its L_1 outputs in (4.3). One vertex for each t_1 checks all its L_2 outputs in (4.4). A vertex may compute this small dense transform by schoolbook arithmetic internally. It has at most $2\max(L_1, L_2) w_A$ ports. These computations are polynomial-time predicates, rather than additional guessed wires. This is precisely the succinct-predicate allowance in Assumption 1.1.

A fully recursive FFT would introduce $\Theta(\log L)$ arrays of wires. We do not use that circuit. The single split above introduces one intermediate array and keeps the degree small enough.

4.4 The actual constraint network

Precompute the transform of the public a. Its L transformed words are public constants, taking $O(L w_A) = O(nr)$ bits. The unknown input is u. Construct the following six arrays, each with L words in A:

1. packed u, with its padding fixed;
2. its forward-transform intermediate y;

3. its forward-transform output z;
4. the pointwise products $\widehat{a}_t z_t$;
5. the inverse-transform intermediate;
6. the inverse-transform output C.

Vertices check each transform stage and each pointwise product. A pointwise vertex reads one input and one output word and multiplies by the hardcoded $\widehat{a}_t$. For the inverse, use ω^{-1} and include the public factor L^{-1} in its second-stage vertices.

To recover the ordinary product, use

$$a(X)u(X) = \sum_t X^{tJ} C_t(X).$$

For each ordinary coefficient there are at most two contributing block coefficients, because $0 \le \deg C_t < 2J$. Folding the coefficients in degrees $n, \dots, 2n-2$ with a minus sign gives the product modulo $X^n + 1$. Therefore each negacyclic output coefficient reads at most four scalar coefficients from C. A final vertex checks the corresponding equation $au + v = c$ or $au - v = c$ modulo 2^r, reading at most $5r$ bits including the coefficient of v.

The nr bits of v connect those vertices to interval-membership predicates. Input vertices enforce binary u (or another stated bounded alphabet) and all packing zeros. For binary u, the nonzero information is its least significant bit; higher bits are fixed to zero. The remaining unused output bits terminate at true predicates. They are not free choices: their values are already forced by their producing transform vertex.

Lemma 4.2 (Correctness and unique extension)**.** *The network accepts an interface pair (u, v) exactly when the stated coefficient bounds and the relation $au \pm v = c$ hold in R_D. For each accepted pair, all auxiliary bits are uniquely determined. For fixed public a, c and u, at most one v is possible.*

Proof. Fix u. The packing constraints fix its entire padded array. Every transform vertex computes a specified function in a unique coefficient representation, and every product vertex does the same. Following the layers thus determines every subsequent array. Lemma 4.1 and (4.2) identify the inverse outputs with the genuine block convolution. The no-wraparound bounds identify their recombination with the ordinary product, and negacyclic folding identifies the result with au in R_D. The final affine equation uniquely fixes v. Membership predicates accept exactly the stipulated coefficients. Conversely a true relation supplies these computed arrays and satisfies every predicate. The forward determination proves uniqueness, including all padded or unused words. □

4.5 Boolean edge count, degree, descriptions, and space

Proposition 4.3 (Explicit encoding bounds)**.** *For public $a, c \in R_D$, the relation $au \pm v = c$ with binary or interval-restricted interface coefficients has an encoding with*

$$E \le 325nr, \qquad \Delta \le 63n^{3/4}r. \tag{4.5}$$

It has polynomial total description length, is constructible in polynomial time, and all its predicates are uniformly polynomial-time evaluable.

Proof. The six arrays and final v edges use

$$E = 6Lw_A + nr = 12LJr + nr.$$

Since $J^2 < 9n$ and $L \le 3J$, this is less than $325nr$. No word is shared by more than two endpoints; transform outputs are permuted to the appropriate next-stage inputs.

Ring-element arrays are bundles of their scalar bit edges, not single edges. Input and range vertices use the other endpoints of the interface bundles.

The largest transform degree is at most

$$2w_A \max(L_1, L_2) \le 4Jr\sqrt{3L} \le 12J^{3/2}r < 63n^{3/4}r.$$

Pointwise degrees are at most $2w_A$, input degrees at most w_A, and final affine degrees at most $5r$, all within this bound.

Each coefficient is an r-bit integer. Direct arithmetic on length-$2J$ polynomials over $\mathbb{Z}/2^r\mathbb{Z}$, followed by monic reduction, takes polynomial time and space in n, r. A small transform has polynomially many such operations. Its description is an algorithm and public parameters, not a table of 2^{Δ} entries. The precomputed public transform is polynomial in size. The graph has $O(n + L)$ vertices. These observations prove all description, construction, and evaluation assertions with fixed polynomial exponents, uniformly over the inputs. □

For m samples, duplicate the convolution network and connect the copies of each secret bit by equality trees. Their additional size is $O(mn)$. As a conservative generic normalization, subdividing every edge gives a bipartite network with at most $700mnr$ edges, allowing slack for those trees. Its degree is still bounded by a constant times $n^{3/4}r$. If $r \le n^{3/2}$, then

$$n^{3/4}r \le (nr)^{9/10}.$$

For example $s_\star = 1000mnr$ safely accommodates the original theorem's edge, precision, and degree conditions. These constants are bookkeeping bounds, not optimized implementation constants.

4.6 A smaller bipartite encoding for binary inputs

The preceding construction is a conservative reference implementation, valid for both binary and bounded-alphabet interfaces. We now remove unnecessary interfaces for binary inputs and prove smaller constants. The single-split transform arithmetic is unchanged.

First note that Lemma 4.1 only requires an odd power-of-three length L dividing $3J$, not that J itself be a power of three. Its proof groups exponents into triples with ratio Y^{Jh_0}, and thus works for every positive integer J satisfying that divisibility. For $n \ge 8$ choose

$$L = \max\{3^a : L^2 \le 3n/2\}, \quad h = \left\lceil 6n/L^2 \right\rceil + 1, \quad J = hL/3, \quad B = \lceil n/J \rceil . \tag{4.6}$$

Here $L \ge 3$, so J is an integer. We have

$$n/6 < L^2 \le 3n/2, \qquad 6n < hL^2 \le 6n + 2L^2 \le 9n.$$

In particular $n/J = 3n/(hL) < L/2$, and since L is odd, $B \le (L+1)/2$. Therefore $2B - 1 \le L$ and both no-wraparound conditions remain valid. The cases $n = 2, 4$ may use the reference network.

Interfaces that can be removed. Each stage-one forward predicate reads the binary input coefficients directly, treating the packing zeros and high coefficient bits as public constants. There are n input edges rather than a padded array of words. Fuse public pointwise multiplication by $\widehat{a}$ into each first-stage inverse predicate; the product array is no longer guessed on edges. Only the $2B - 1$ potentially nonzero inverse output words C_t are emitted. Finally, an interval-only fold predicate reads at most four coefficients and checks $c_i - (au)_i \in I \pmod D$ internally; no separate error word is needed. For either sign in $au \pm v = c$, v is determined internally.

There remain three full arrays (forward intermediate, forward output, inverse intermediate), the cropped inverse output, and n input bits. They form successive layers

$$\text{input} \to \text{forward } 1 \to \text{forward } 2 \to \text{inverse } 1 \to \text{inverse } 2 \to \text{fold}.$$

These arrows describe functional determination, not directed graph edges. Alternating the vertex colors proves that the graph is already bipartite. Unused output coefficients terminate at true vertices of the fold color. Every emitted word remains a uniquely determined function of the input; omitted inverse words are identically zero for a genuine padded product. Thus the correctness and unique-extension proof of Lemma 4.2 still applies after eliminating these deterministic interfaces.

Proposition 4.4 (Proved compact binary interval-test bounds)**.** *For $n \geq 8$, the binary-input, interval-residual network above is bipartite and has*

$$E = (3L + 2B - 1)\, 2Jr + n \leq 24nr + n, \qquad \Delta \leq 23n^{3/4}r. \tag{4.7}$$

Descriptions, construction, and predicate evaluation remain polynomial.

Proof. The edge formula counts the four retained array interfaces and the secret bits. Using $2B - 1 \leq L$ and $J = hL/3$ gives $E - n \leq 8LJr = (8/3)hL^2r \leq 24nr$. The largest transform predicate has at most $4Jr \max(L_1, L_2) \leq 4Jr\sqrt{3L}$ ports. The direct binary inputs only lower the old input-port count; folding needs at most $4r$ ports.

To prove the constant uniformly, put $x = L/\sqrt{n}$. Since $J \leq 2n/L + 2L/3$, the transform degree divided by $n^{3/4}r$ is at most

$$f(x) = 4\sqrt{3}\left(\frac{2}{\sqrt{x}} + \frac{2}{3}x^{3/2}\right), \qquad 1/\sqrt{6} < x \leq \sqrt{3/2}.$$

Its derivative has the sign of $x^2 - 1$, so its maximum is at an endpoint. At the lower endpoint its value is $(76/9)\sqrt{3}\,6^{1/4} < 23$, and at the upper endpoint it is $12\sqrt{3}/(3/2)^{1/4} < 23$. This proves the degree bound without extrapolating finite tests. The remaining assertions follow from the same polynomial arithmetic and canonical representations as before. □

The bound $22nr + n$ can be observed in finite parameter searches, but is not used as a proved universal constant here. Formula (4.7) permits exact counts for each selected (L, J); other choices may be optimized subject to all degree and root conditions.

Many samples and equality degrees. Using one m-port equality vertex per secret bit gives mE edges, but its maximum degree is $\max\{\Delta, m\}$. The m must not be omitted from a degree certificate. Alternatively use a trivalent equality tree with m external edges. Its internal vertices can all have the input color by subdividing each internal tree edge once with an equality vertex. For $m \geq 3$ there are $m - 3$ internal tree edges; $m = 1, 2$ use a true endpoint or a single equality vertex. This adds at most $2mn$ edges, keeps degree at most $\max\{\Delta, 3\}$, and has exactly one extension for each choice of the common secret bit. Thus for all m one may use

$$E_m \leq 24mnr + 3mn. \tag{4.8}$$

Explicit solver certificates and their domains. For an already bipartite network use $s_\star = \max\{E_m, \lceil \Delta^{10/9} \rceil, 2\}$; no edge subdivision is needed. Counting precision $P = E_m + 1 \leq 2s_\star$ is linear in the size parameter, as permitted by the original source theorem. For $r \geq 2$ and $r \leq n^{3/2}$,

$$\Delta^{10/9} \leq 23^{10/9}nr < 33nr, \qquad E_m \leq 25.5mnr,$$

so $s_\star \le 33mnr$ is a safe integer certificate. In the narrower range

$$r \le n^{3/2}/1000, \tag{4.9}$$

the degree term is at most $23^{10/9}nr/1000^{1/9} < 16nr$. Then $26mnr$ suffices with the bounded-degree equality trees. For one sample, $E \le 24.5nr$ and $25nr$ suffices. The same $25mnr$ certificate applies to the direct m-port construction when its additional degree satisfies $m^{10} \le (25mnr)^9$. These are proved scope conditions, not a uniform $25mnr$ claim over the original full range of r, m.

Gaussian norm tests need additional interfaces. A norm test cannot eliminate the residual words at the fold. Emit their nr bits, then n centered squares with $2r$ bits each, and a balanced sum tree. If $k = \log_2 n$, the sum-output edges total

$$\sum_{j=1}^{k} \frac{n}{2^j}(2r + j) = 2r(n-1) + 2n - k - 2.$$

Including residuals and squares adds fewer than $5nr + 2n$ edges to the interval-only count. The tree layers preserve bipartiteness and unique extension. Their degree is at most $6r + 3\log_2 n$, which fits the $23n^{3/4}r$ bound for $n \ge 8, r \ge 2$. For m separate norm tests the resulting bound is $E_m \le 29mnr + 5mn \le 31.5mnr$. Thus $33mnr$ remains sufficient in the full range $r \le n^{3/2}$, and $32mnr$ suffices in (4.9). A $25mnr$ interval-test certificate must not be reused without this accounting.

Other input alphabets and optional narrower forward words. The n-bit input interface assumes an exactly binary secret. An alphabet of L_s consecutive integers needs at least the corresponding encoded coefficient words, not one bit per coefficient. The conservative network already proves its required $O(nr)$ bound when that alphabet fits modulo D; the binary constants above are not asserted unchanged.

For a binary input one can also carry forward-transform coefficients as signed integers rather than full r-bit residues. After reduction of a monomial modulo $Y^{2J} + Y^J + 1$, it contributes at most two signed monomials, each with coefficient of absolute value one. Each input bit therefore contributes at most one to a given output coefficient. Both the first-stage partial transform (including its twiddle) and the full transform have coefficient magnitudes at most n. Representing them with $\lceil \log_2(2n+1) \rceil$ signed bits and evaluating the inverse input modulo D is exact and preserves unique extension. It can help when r is much larger than $\log n$, but a bounded nonbinary input adds its coefficient magnitude to this bound. In particular an error-first input of width W needs an extra $O(\log W)$ bits; the binary estimate cannot be transferred to it verbatim. We do not use this optional optimization in any displayed uniform certificate.

Remark 4.5 (What the construction does not hide). There are no exponentially large public lists, no approximate complex arithmetic, no unrecorded intermediate arrays, and no rounding operations. The only internal computations not expanded into wires occur inside polynomial-time predicates, which are expressly allowed by the assumed theorem. Explicit truth-table tensors or constant-fan-in Boolean circuits would be a different model; our linear edge bound does not claim linear size in either of those models.

5 Overall conditional complexity and parameter thresholds

Theorem 5.1 (Binary Ring-LWE attack). *Assume Assumption 1.1. Let $n = 2^k \ge 2$, $q = 2^t$, $D = 2^r \mid q$, $W < D$, $m \ge 1$, and $r \le n^{3/2}$. Algorithm 1 returns the true binary*

secret or reports failure. Failure is at most the exact first moment (3.3), and hence also at most the coarse bound $[2(\alpha_W/D)^m]^n$. *Its time is*

$$\mathrm{poly}(n, m, t) + 2^{O\left(mnr/\sqrt{\log(mnr)}\right)}, \tag{5.1}$$

and its space is polynomial in n, m, t. *The assertion holds for every fixed binary secret. All constants in the exponent are independent of* n, m, r, t *under the stated uniform encoding bounds.*

Proof. Exact reduction costs polynomial time in the input bit length $O(mnt)$. Proposition 4.3 and the following multi-sample construction give a permitted solver parameter $s_\star = O(mnr)$ and polynomial description length. Assumption 1.1 therefore counts the candidates within the stated exponential bound and polynomial space. Lemma 1.2 adds at most n queries, a polynomial factor absorbed in the exponent for growing mnr. The true secret always belongs to the candidate set. Lemma 4.2 makes the network parsimonious, so a count of one certifies that the returned secret is the true one. Theorem 3.3 bounds the probability of a larger count, with the sharper first moment supplied by Theorem 3.6. Only one query and its work space are retained at a time. □

For binary interval tests with $n \geq 8$, Section 4.6 proves $s_\star \leq 33mnr$ over the theorem's full range, and smaller certificates in (4.9). These improve bookkeeping constants, not the asymptotic exponent. The conservative network remains available for the small dimensions and other interfaces.

5.1 Subexponential in which parameter?

Here "subexponential in n" means $2^{o(n)}$, with polynomial input overhead. It does not mean subexponential in the numerical value of the modulus, or merely subexponential in the total bit length of an arbitrarily expanded verifier. Assume t and the input length are polynomial in n. Put $R = mr$. The derived exponential term is subexponential in n if

$$R = o(\sqrt{\log n}). \tag{5.2}$$

In fact this is exactly the condition for the expression $nR/\sqrt{\log(nR)}$ itself to be $o(n)$. To see the reverse direction, square the ratio: if it tends to zero, then $R^2 = o(\log n + \log R)$. For unbounded R, $\log R = o(R^2)$, so necessarily $R^2 = o(\log n)$. Bounded R satisfies the condition automatically. This is a statement about our bound, not a lower bound against every possible attack.

When $q \geq 4W$, Corollary 3.4 permits $m = 1$ and $r = \lceil \log W \rceil + 2$. Thus

$$\log(W + 1) = o(\sqrt{\log n}) \quad \Longrightarrow \quad T = 2^{o(n)}. \tag{5.3}$$

For $n = 2^k$, this reads $\log W = o(\sqrt{k})$.

Width family (sufficient modulus available)	Derived exponential term
Fixed W	$2^{O(n/\sqrt{\log n})}$
$W = (\log n)^a$, fixed a	$2^{O(n \log\log n/\sqrt{\log n})}$
$W = 2^{a\sqrt{\log n}}$, fixed $a > 0$	$2^{O(an)}$
$W = n^a$, fixed $a > 0$	$2^{O(an\sqrt{\log n})}$

The third row is potentially a better single-exponential algorithm, but its leading constant is not given by a big-O statement. The fourth row does not establish an advantage over 2^{cn} attacks.

5.2 Joint choice of divisor and sample count

For any fixed reliability margin $\eta > 0$, one can choose

$$m = \left\lceil \frac{1+\eta}{r - \log \alpha_W} \right\rceil, \qquad \log W < r \le t. \tag{5.4}$$

Then the ambiguity probability is at most $2^{-\eta n}$. The size to optimize is mr, not r alone. If q is well above W, one sample at approximately $\log W + 2$ bits is a simple efficient choice. If q/W is close to one, more samples may be required and the factor

$$mr \text{ is roughly } \frac{(1+\eta)\log q}{\log(q/\alpha_W)}$$

can be large. The exact first moment may improve this conservative choice, but its m-sample expression must be optimized with the powers $A_j(\epsilon_W)^m$ intact. Increasing n does not remove this sample-information requirement. The exact minimization is over integers $r \le t$ using (5.4). $W = q$ is excluded because it supplies no information at all.

5.3 The 2^{64} example and what it does not estimate

For centered interval noise $|e_i| < 2^B$, its width is $W = 2^{B+1} - 1$. Corollary 3.4 gives $r = B + 3$.

Absolute bound	W	retained r	schematic r^2
2^{20}	$2^{21} - 1$	23	529
2^{24}	$2^{25} - 1$	27	729
2^{30}	$2^{31} - 1$	33	1089
2^{40}	$2^{41} - 1$	43	1849

The strict absolute bound is intentional; including an endpoint changes the integer rounding but not the asymptotics. These are retained-modulus and identifiability calculations, not bit-security estimates. A Gaussian standard deviation of 2^{20} is not a hard 20-bit bound. Section 12 treats that distinction.

For fixed $q = 2^{64}$ and fixed $W < q$, all permitted m, r are constants as $n \to \infty$. The theorem then has a subexponential asymptotic term. This observation proves subexponentiality of our own conditional bound, not superiority over every conventional algorithm. Sections 6.6 and 6.7 compare with original-modulus attacks and derive the fixed-modulus behavior of an explicit primal model. The unknown constants and solver cutoff must still be applied before asserting any attainable crossover dimension.

5.4 Verification, success probability, and alternative attacks

The reduced search uses at most the selected m ring samples; additional public samples are not implicitly supplied for free. The true secret's membership is exact for bounded noise. An implementation can count uniqueness and return failure, giving no wrong output under that promise. For a noisy distribution handled by typical sets instead, the true vector can fall outside the acceptance set with a separately bounded probability.

One should compare sample complexity as well as running time. With very many samples, bounded-error algebraic attacks can exploit low-degree polynomial relations; a prototype is linearization of

$$\prod_{e \in I} (b - \langle a, s \rangle - e) = 0.$$

This is the starting point of Arora–Ge-type methods [10]. Their applicability and rank guarantees depend on the coefficient ring and sample distribution. We make no claim of

a globally best algorithm in an unlimited-sample model. Our specific contribution is a one/few-ring-sample reduction, compared in the next sections with the standard BKZ route.

6 Where the attack sits relative to BKZ and sieving

6.1 Lattices, projections, and the SVP subproblem

A basis matrix $B = (b_1, \dots, b_d)$ with linearly independent columns generates $\mathcal{L}(B) = \{Bz : z \in \mathbb{Z}^d\}$. Its covolume is $\det(\mathcal{L}) = \sqrt{\det(B^T B)}$; for square B it is $|\det B|$. Gram–Schmidt orthogonalization writes

$$b_i = b_i^* + \sum_{j<i} \mu_{i,j} b_j^*, \qquad \mu_{i,j} = \frac{\langle b_i, b_j^* \rangle}{\|b_j^*\|^2}.$$

The triangular change of basis has diagonal one, so the volume is $\prod_i \|b_i^*\|$. Let π_i be orthogonal projection onto the complement of $\mathrm{span}_{\mathbb{R}}(b_1, \dots, b_{i-1})$. For a block starting at i with length $h \le \beta$, the relevant block lattice is generated by

$$\pi_i(b_i), \dots, \pi_i(b_{i+h-1}).$$

The shortest-vector problem (SVP) asks for a nonzero lattice vector of least Euclidean norm. The block's covolume in its own span is $\prod_{j=i}^{i+h-1} \|b_j^*\|$, by the same triangular-volume argument.

BKZ repeatedly uses an SVP procedure on such projected blocks and inserts short combinations when they improve the basis. If $v = \sum_{j=i}^{i+h-1} z_j \pi_i(b_j)$ is a shortest vector, its coefficient vector can be chosen primitive: if all z_j had common divisor greater than one, division by that divisor would give a shorter nonzero block vector. A primitive integer vector extends to a unimodular matrix (successive Euclidean algorithms reduce its coordinates to their gcd). It can therefore be inserted by integer basis operations without changing the lattice. Size reduction and Gram–Schmidt updates follow. Blocks are processed in tours until a chosen stopping rule or quality criterion is met.

This explains the role of the block size: larger β makes each local search harder but typically improves the basis quality. Modern schedules include preprocessing, pruning, early termination, and changing block sizes; Chen–Nguyen's BKZ 2.0 is a central reference [5]. None of those choices is fixed by saying only "run BKZ."

6.2 Why sieving appears inside the cost estimate

A lattice sieve maintains many lattice vectors. Combining two nearby vectors can shorten them: if $\|x\| = \|y\| = R$, then

$$\|x - y\|^2 = 2R^2 - 2\langle x, y \rangle < R^2 \quad \Longleftrightarrow \quad \langle x, y \rangle > R^2/2.$$

Thus vectors at angle below 60 degrees are useful reducers. The search for such pairs is accelerated by near-neighbor data structures rather than testing every pair.

The familiar list-size estimate can be motivated geometrically. For a uniform point on the unit sphere in dimension h, the density of one coordinate is proportional to $(1 - z^2)^{(h-3)/2}$. Integrating above $z = 1/2$ gives an exponential factor $(3/4)^{h/2+o(h)}$ for the corresponding spherical cap; normalization and endpoint widths contribute subexponential factors. An inverse-cap-sized list is therefore roughly $(4/3)^{h/2+o(h)} = 2^{0.2075\ldots h+o(h)}$. This calculation by itself is not a proof that a lattice sieve succeeds: it invokes the heuristic that the normalized lattice vectors behave like independent uniform sphere points.

The BDGL locality-sensitive-filtering analysis gives the conventional *heuristic benchmark*

$$C_{\mathrm{SVP}}(h) = 2^{(c_{\mathrm{svp}}+o(1))h}, \qquad c_{\mathrm{svp}} = \tfrac{1}{2}\log_2(3/2) = 0.292481\ldots. \tag{6.1}$$

We use (6.1) as a comparison model, not as a theorem proved by our reduction. The original analysis and its time–space tradeoffs are in [6]. The list alone is exponential space; extra filter storage and the precise sieve variant affect total space. Enumeration-based block solvers offer different time–space behavior. Polynomial space for the conditional tensor algorithm therefore should not be confused with an exponential-space lower bound for every SVP algorithm.

6.3 The standard primal lattice embedding

Write the selected ℓ scalar equations from the Ring-LWE samples as

$$As + e = b \pmod q, \qquad A \in \mathbb{Z}_q^{\ell\times n}.$$

For complete ring samples, A consists of negacyclic multiplication matrices. A subset of rows can also be used. The distinction affects the statistics but not the following exact embedding. Define the rank-d lattice, $d = n + \ell + 1$,

$$\mathcal{L}_{\mathrm{emb}} = \{(u, v, z) \in \mathbb{Z}^n \times \mathbb{Z}^\ell \times \mathbb{Z} : Au + v - zb = 0 \pmod q\}. \tag{6.2}$$

A basis is

$$B_{\mathrm{emb}} = \begin{pmatrix} I_n & 0 & 0 \\ -A & qI_\ell & b \\ 0 & 0 & 1 \end{pmatrix}. \tag{6.3}$$

Every displayed column satisfies the congruence. Conversely, for any (u, v, z) in (6.2), the vector $(v + Au - zb)/q$ is integral and expresses it in these columns. Thus the basis is exact, with determinant q^ℓ by its block-triangular form. The planted vector $(s, e, 1)$ belongs to it.

To balance differently sized secret and error coordinates, apply a public diagonal scaling with positive rational weights w_s, w_e, w_z. Rational denominators can be cleared globally. The volume becomes

$$\Delta = q^\ell w_s^n w_e^\ell w_z,$$

and the planted squared norm is

$$\tau^2 = w_s^2 \|s\|^2 + w_e^2 \|e\|^2 + w_z^2. \tag{6.4}$$

For binary secrets $\|s\|^2 \le n$. After subtracting a known integer interval center from the labels, $|e_i| \le C$ gives $\tau^2 \le nw_s^2 + \ell C^2 w_e^2 + w_z^2$. For a centered uniform interval the variance is $(W^2 - 1)/12$, obtained by summing consecutive squares; when its center is a half-integer, a known half-integer mean can instead be handled by a global scaling.

The usual attack hopes that sufficiently strong reduction finds this planted vector, or enough related information to recover it. A short vector need not be the planted one, and a unique feasible binary secret does not imply that $(s, e, 1)$ is the shortest vector of the embedding. For example, vectors with $z = 0$ are unrelated homogeneous relations. Verification of the candidate's secret and errors is essential.

6.4 The common heuristic quality model, with its assumptions exposed

The Gaussian heuristic estimates the shortest-vector scale of a generic h-dimensional lattice of volume V by the radius of a ball of volume V:

$$\mathrm{GH}(h, V) = (V/\omega_h)^{1/h}, \qquad \omega_h = \frac{\pi^{h/2}}{\Gamma(h/2 + 1)}.$$

The formula for ω_h follows from integrating a Gaussian in Cartesian and polar coordinates. Stirling's formula gives $\omega_h^{-1/h} = \sqrt{h/(2\pi e)}(1 + o(1))$. Treating this volume scale as an actual SVP scale is a heuristic; no such step occurs in the tensor attack.

Another heuristic, the geometric-series assumption (GSA), models Gram–Schmidt lengths after reduction as

$$g_i = \delta_\beta^{d-2i+1} \Delta^{1/d}, \qquad 1 \le i \le d. \tag{6.5}$$

The exponents sum to zero, so this model has the correct volume. In a β-block its determinant is $g_i^\beta \delta_\beta^{-\beta(\beta-1)}$. Equating its Gaussian-heuristic shortest scale to g_i yields

$$\delta_\beta^{\beta-1} \approx \omega_\beta^{-1/\beta}, \qquad \delta_\beta \approx (\beta/(2\pi e))^{1/(2(\beta-1))}.$$

This derivation explains the model rather than proving it describes actual BKZ outputs.

If the target has a generic orientation, projecting to a random β-dimensional subspace retains squared norm β/d in expectation. Indeed rotational symmetry makes the expected squared component equal in each orthogonal coordinate and their sum is τ^2. Treating a particular BKZ tail projection as having norm $\sqrt{\beta/d}\,\tau$ is again a heuristic. The last block starts at $i = d - \beta + 1$, so (6.5) suggests the approximate success condition

$$\sqrt{\beta/d}\,\tau \lesssim \delta_\beta^{2\beta-d-1} \Delta^{1/d}. \tag{6.6}$$

This is an explanatory primal model, not a certified security estimator. Uniqueness gaps, deviations from GSA, progressive schedules, hybrid guessing, and sample selection can all change the optimum. In particular, the noise scale, ℓ, and the weights affect the required β; it cannot be replaced by n by definition.

6.5 Overall cost accounting and the correct comparison

For any specified BKZ schedule, let its successive SVP dimensions be $h_1, \dots, h_N$. Its accounting identity is

$$T_{\mathrm{BKZ}} = T_{\mathrm{setup}} + T_{\mathrm{updates}} + \sum_{j=1}^{N} C_{\mathrm{SVP}}(h_j). \tag{6.7}$$

Preprocessing calls must be included in the sum; they are not free. If the schedule has polynomially many calls, all of size at most β, and uses (6.1), its leading heuristic exponent is at most $c_{\mathrm{svp}}\beta + o(\beta)$, up to the logarithm of the call count. The bound is attained within this model when the schedule includes a full β-dimensional call and polynomial setup and updates do not dominate. If the number of calls is not polynomial, that simplification is not available. This is the reason cryptanalytic estimates normally report both a required block size and an SVP cost model.

For an instance in our structured family, the two routes are alternatives:

$$\begin{aligned} T_{\text{ring attack}} \;\le\; \min\{ & 2^n \,\mathrm{poly}(n, m, t), \\ & T_{\mathrm{direct}}(n, m, r), \\ & T_{\mathrm{BKZ}}(d, \beta, \text{schedule}), \\ & T_{\text{other applicable attacks}}\}. \end{aligned}$$

The direct term is the conditional bound of Theorem 5.1. It acts on the original public ring equations and bypasses the embedding, Gram–Schmidt computation, BKZ tours, and sieve lists. It can replace the *whole high-level recovery route* for these instances. It has not been proved to replace the SVP subroutine in (6.7).

The conventional term refers to the original modulus q and all samples actually available. The next two subsections make the modulus choice and the asymptotic qualification explicit. We do not prove a lower bound for BKZ or claim optimality among all attacks.

6.6 Original versus reduced modulus: which instance is the benchmark?

The tensor route deliberately reduces to $D = 2^r \mid q$. A conventional attacker is not obliged to discard the other bits. Define

$$T^{\star}_{\mathrm{conv}}(n, m, q, I) = \inf_{\mathcal{A} \in \mathcal{C}} T_{\mathcal{A}}(n, m, q, I),$$

where $\mathcal{C}$ is a stated collection of conventional attack models, each meeting the same prescribed recovery probability. Its optimization may include sample selection, weights, primal and dual attacks, hybrid guessing, and useful preprocessing. No assertion that this infimum is known, attained, or a lower bound on algorithms outside $\mathcal{C}$ is intended. Sample reading and preprocessing are part of each cost. The comparison of interest is

$$T_{\mathrm{direct}}(n, m, r, I) \quad \text{versus} \quad T^{\star}_{\mathrm{conv}}(n, m, q, I). \tag{6.8}$$

For m ring samples there are mn scalar rows available; the m in our network size must not be entered as the scalar sample count in an LWE estimator. Those rows also retain the ring correlations already noted.

There is an exact algorithmic reason for allowing the original modulus. Compose any reduced-instance binary-secret algorithm with coefficientwise reduction of the original input. Lemma 2.5 proves that its input distribution is exactly the required reduced distribution. Its binary output is already the full secret. Thus every reduced-instance algorithm is an original-instance algorithm with polynomial preprocessing overhead. For a class closed under this composition, its optimal original-instance cost cannot exceed its optimal reduced-instance cost, up to that overhead. This statement does not require a monotonicity theorem for estimator output.

For the fixed primal embedding, one can see the effect of discarding high bits directly. Keep the selected ℓ rows, weights, secret, and integer errors fixed. The target norm τ is unchanged, whereas the volume changes from Δ_q to

$$\Delta_D = (D/q)^{\ell} \Delta_q.$$

Consequently the target-to-volume ratio changes by the exact factor

$$\frac{\tau/\Delta_D^{1/d}}{\tau/\Delta_q^{1/d}} = (q/D)^{\ell/d}. \tag{6.9}$$

Within a Gaussian-heuristic interpretation the target is less exceptional after reduction. Its absolute noise has not decreased, and its relative noise has increased by q/D. This explains why a conventional estimate can become worse at D even though the tensor encoding becomes shorter. Reoptimization can alter the numerical effect; (6.9) holds for the fixed embedding, not as a theorem about every attack.

6.7 Fixed modulus and noise: what the primal model actually predicts

We now substantiate the linear-block-size premise *inside the specific model* (6.6). This is useful for the intended asymptotic family: fixed $q = 2^{64}$, fixed interval noise, fixed number of ring samples, and $n \to \infty$. It is not a claim that BKZ has a proved exponential lower bound. Write ln for natural logarithms in this calculation.

Proposition 6.1 (Fixed-modulus consequence of the primal GSA model)**.** *Suppose q is fixed, $d = n + \ell + 1 \to \infty$, and the target satisfies*

$$\|s\|^2 \geq c_s n, \qquad \|e\|^2 \geq c_e \ell,$$

for constants $c_s, c_e > 0$. Allow arbitrary positive weights, even depending on n. Interpret the approximate success test as the inequality

$$\sqrt{\beta/d}\,\tau \le C_0 \delta_\beta^{2\beta-d-1} \Delta^{1/d}, \tag{6.10}$$

where C_0 is a fixed constant. Assume $\beta \le d$, that every allowed fixed block size has $\delta_\beta > 1$, and that at growing block sizes

$$\ln \delta_\beta = \frac{\ln \beta + O(1)}{2\beta}.$$

Then every sequence satisfying this model test has $\beta/d \to 1$. In particular, for fixed m and $\ell \le mn$ it has $\beta = \Theta(n)$.

Proof. Put $c = \min\{c_s, c_e, 1\} > 0$. The target norm and weighted arithmetic– geometric mean inequality give

$$\tau^2 \ge c(n w_s^2 + \ell w_e^2 + w_z^2), \qquad (w_s^n w_e^\ell w_z)^{1/d} \le \sqrt{\frac{n w_s^2 + \ell w_e^2 + w_z^2}{d}}.$$

Since $q^{\ell/d} \le q$, it follows that $\Delta^{1/d} \le q\tau/\sqrt{cd}$. Cancel $\tau/\sqrt{d}$ in (6.10) to obtain

$$\sqrt{\beta} \le C \delta_\beta^{2\beta-d-1}, \qquad C = C_0 q/\sqrt{c}. \tag{6.11}$$

A bounded subsequence of block sizes has a fixed value on a further subsequence. Its right side tends to zero because $\delta_\beta > 1$, a contradiction. Thus $\beta \to \infty$. If $\beta \le d/2$ infinitely often, the right side is at most C there, another contradiction. Hence eventually $d/2 < \beta \le d$.

Taking logarithms in (6.11), and using that $(2\beta - d - 1)/\beta$ is now bounded, gives

$$\tfrac{1}{2} \ln \beta \le \ln C + \left(1 - \frac{d}{2\beta}\right) \ln \beta + O(1).$$

Rearranging yields

$$0 \le \left(\frac{d}{\beta} - 1\right) \ln \beta \le O(1).$$

Therefore $d/\beta \to 1$. With fixed m, the bounds $n+1 \le d \le (m+1)n+1$ imply $\beta = \Theta(n)$. □

For a uniform binary secret, its squared norm is a sum of n independent Bernoulli variables and exceeds $n/4$ except with exponentially small probability. Independent fixed-width interval errors of nonzero second moment similarly have squared norm exceeding a fixed positive multiple of ℓ, except with exponentially small probability when $\ell \to \infty$. For completeness, if bounded nonnegative Z has mean $\mu > 0$, choose $0 < a < \mu$. The function $g(\theta) = \ln \mathbb{E} e^{\theta(a-Z)}$ has $g(0) = 0$ and $g'(0) = a - \mu < 0$, so some fixed $\theta > 0$ gives $g(\theta) < 0$. Markov's inequality then bounds $\Pr[\sum_{i=1}^N Z_i \le Na] \le e^{Ng(\theta)}$. Apply this to squared secret and error coefficients. These observations justify the energy hypotheses for typical inputs and a fixed selection of rows. They do not cover arbitrary label-dependent selection or every fixed binary secret, including the trivial all-zero secret. That distinction concerns this BKZ comparison, not the recovery theorem.

Under the model and polynomially many calls including a full block, the proposition gives the exponential sieve benchmark $2^{\Theta(n)}$. Our fixed-m, r tensor bound is $2^{O(n/\sqrt{\log n})}$, so it eventually beats *that benchmark*. The same argument uses the original fixed q, not D. A large constant q affects C and the finite range but not this model's limiting linear block-size conclusion.

One can also check that the prediction is nonvacuous in a simple instance of the model. Take one full ring sample, $\ell = n$, a uniform binary secret, and centered uniform noise of odd width $W \geq 3$, with $\sigma^2 = (W^2 - 1)/12$. Choose $w_s = \sqrt{2}$, $w_e = 1/\sigma$, and $w_z = 1$ for this model calculation; rational approximations with strict slack give the same limiting conclusion. Then $\tau/\sqrt{d} \to 1$ in probability, while $\Delta^{1/d} \to \sqrt{q\sqrt{2}/\sigma}$. With the explicit approximation $\delta_d = (d/(2\pi e))^{1/(2(d-1))}$ and $\beta = d$, the ratio of the right side of the model test with $C_0 = 1$ to its left side tends to

$$\sqrt{\frac{q\sqrt{2}}{2\pi e\,\sigma}}.$$

It exceeds one when $q/\sigma > 2\pi e/\sqrt{2}$. In particular $q \geq 4W$ implies $q/\sigma > 4\sqrt{12} > 2\pi e/\sqrt{2}$. Thus the full block satisfies this idealized test with a fixed margin. The following qualification still applies: passing that test does not prove actual recovery by SVP.

Why this remains a model calculation. The embedding has substantial nonrandom structure at fixed q and unbounded dimension. It contains $(0, qe_i, 0)$ of weighted norm qw_e. But the target has norm at least $w_e\sqrt{c_e\ell}$, so for $\ell > q^2/c_e$ these homogeneous vectors are shorter than the planted target, for *every* choice of weights. Thus even exact full-dimensional SVP need not return the target. Unique binary feasibility does not remove these vectors. Actual q-ary profiles, homogeneous-vector handling, decoding, and attack variants require additional analysis; satisfying the displayed model inequality is not sufficient to prove recovery. This elementary obstruction also shows why a global random-lattice or GSA picture cannot simply be treated as a proved uniform approximation at all asymptotic dimensions. Proposition 6.1 proves an implication of its explicit model assumptions, not a lower bound for all BKZ-based attacks. We have not exhibited a subexponential BKZ attack for the fixed-q, fixed-noise family, either.

The Lattice Estimator documentation itself distinguishes the basic GSA calculation, profile simulation, BDD, and hybrid primal attacks. Its ZGSA profile explicitly retains q-vectors [12, 13]. Our proposition analyzes the displayed model here; it is not a reported run of that software or a claim that every estimator mode has its limit.

If an attack guesses g secret bits before reduction, the same argument applies to a residual model satisfying the hypotheses. Guessing $g = \Omega(n)$ costs $2^{\Omega(n)}$ for ordinary exhaustive guessing, while $g = o(n)$ leaves a residual dimension of order n. This observation does not analyze more sophisticated hybrids, dual methods, or every possible preprocessing transformation. Those belong in the original-instance optimization (6.8).

6.8 Why a block cannot simply be called a smaller Ring-LWE instance

Two exact obstructions suffice. First, projected block lattices are not generally closed under ring multiplication. In dimension two the ring rotation is $J(x, y) = (-y, x)$. Projecting the ideal lattice $\mathbb{Z}^2$ onto the complement of $(1, 1)$ gives a one-dimensional lattice on $\mathbb{R}(1, -1)$, but $J(1, -1) = (1, 1)$ is not on that line. Thus even this simple projection does not inherit the original ring action.

Second, a binary coefficient promise is not invariant under unimodular basis changes. For $U = \left(\begin{smallmatrix} 1 & M \\ 0 & 1 \end{smallmatrix}\right)$, the same coefficient vector is written in the new basis as $U^{-1}s = (s_1 - Ms_2, s_2)$. A binary s may therefore have a coefficient of size M. A generic SVP call receives neither a known binary target in the original power basis nor public equations of the form used by our network. Any proposed BKZ acceleration must supply and analyze that missing reduction, including its size and approximation loss. The present paper does not assume it exists.

7 The cutoff and the crossover against a BKZ model

7.1 An asymptotic rate is not a numerical running-time estimate

Subexponentiality concerns a family as $n \to \infty$. It does not specify a finite point where a particular implementation becomes faster. To express what would be needed, write a quantitative version of the source bound as

$$T_{\mathrm{count}}(s) \le s^B 2^{Ks/\sqrt{\log s}} \qquad (s \ge S_0), \tag{7.1}$$

with constants K, B, S_0 for the permitted input family. The theorem as quoted does not give practical numerical values for all three. Let A account for our encoding and the size parameter, so $s \le Amnr$. The reference construction permits $A = 1000$. For binary interval tests, Section 4.6 proves $A = 33$ when $n \ge 8$ throughout $2 \le r \le n^{3/2}$, $A = 26$ in the narrower range (4.9), and $A = 25$ there for one sample (or when its stated equality-degree condition holds). Norm tests have their own 33 or 32 certificates. Choose the certificate for the interface actually used; a bound from interval-only binary search is not automatically valid for every extension. Self-reduction and input work give a bound of the form

$$\log_2 T_{\mathrm{direct}} \le \frac{KA\,mnr}{\sqrt{\log(Amnr)}} + B' \log(nmt) + O(1), \tag{7.2}$$

for sufficiently large parameters, with a suitable fixed B'. The polynomial input term must also be included if it dominates.

For a chosen conventional attack on the *original modulus* q, let β_q be its maximum block size and N_q its number of SVP calls. Assume its cost model is dominated by calls of that size and the sieve model (6.1); otherwise use its actual modeled cost sum. The comparison is then with $c_{\mathrm{svp}}\beta_q + \log_2 N_q + o(\beta_q)$, not automatically with $c_{\mathrm{svp}}n$, and not with a potentially larger block size β_D for the reduced instance. Thus a sufficient *model crossover* requires

$$Amnr \ge S_0, \tag{7.3}$$

$$\frac{KA\,mnr}{\sqrt{\log(Amnr)}} + B' \log(nmt) < c_{\mathrm{svp}}\beta_q + \log_2 N_q + o(\beta_q). \tag{7.4}$$

This compares an upper estimate with an explicitly selected baseline model. To beat the best model in a chosen collection, compare with $\log_2 T^{\star}_{\mathrm{conv}}(n, m, q, I)$ instead, allowing each model its own optimized preprocessing, sample selection, and weights. If a model also guesses bits, its guessing exponent must be included. This is not a lower bound on the true cost of all conventional attacks.

If $\beta_q = \rho n$ with fixed $\rho > 0$, m, r are fixed, and polynomial terms are negligible, the schematic requirement is

$$\log_2(Amnr) > \left(\frac{KA\,mr}{c_{\mathrm{svp}}\rho}\right)^2. \tag{7.5}$$

For a target $2^{\varepsilon n}$, replace $c_{\mathrm{svp}}\rho$ by ε. This explains the scale $n = 2^{\Theta((mr)^2)}$ only after the constants and the source's starting range are fixed. Merely setting $KA = 1$ is not a security estimate. Replacing β_D by a smaller β_q makes this comparison less favorable to the tensor attack: reducing ρ increases the required logarithmic dimension as $1/\rho^2$ in this schematic formula. It can radically change a crossover without changing the eventual exponential-versus-subexponential distinction.

7.2 Eventual advantage: separate the fixed family from varying parameters

Put $F(n) = mnr/\sqrt{\log(mnr)}$. For a baseline with polynomially many calls, negligible polynomial overhead, and modeled exponent proportional to β_q, the relevant comparison is $F(n)$ versus $\beta_q(n)$. If $F = o(\beta_q)$, the conditional tensor upper bound is eventually smaller than that baseline, regardless of its fixed hidden constant. If $F = \Theta(\beta_q)$, leading constants decide. If $\beta_q = o(F)$, our upper bound establishes no improvement. This last case does not prove that the actual tensor algorithm is slower, since an upper bound is not a lower bound on its running time.

For fixed $q = 2^{64}$, fixed W with $q \geq 4W$, and one ring sample, $r = \lceil \log W \rceil + 2$ is fixed. Hence $F = \Theta(n/\sqrt{\log n})$. Proposition 6.1 derives $\beta_q = \Theta(n)$ within the specified direct primal GSA model, for typical positive-energy inputs and successful sequences of the model test. Consequently the tensor bound eventually beats this exponential sieve benchmark. Making conventional attacks harder by reducing to D is not needed for that conclusion: the model calculation already uses the original fixed q. Its structural limitations were explained alongside the proposition; neither this comparison nor the proposition proves a lower bound on optimized actual BKZ or all conventional attacks. In particular, at the enormous dimensions involved in the source solver's cutoff, validity of a conventional profile model cannot be assumed from its accuracy at cryptographic dimensions. The short homogeneous vectors exhibited in Section 6.7 are an explicit reason to keep these two assertions separate.

As a logical contrast, suppose a different applicable baseline had $\beta_q = \Theta(n/\log n)$ and polynomially many calls. Its sieve model would *itself* be subexponential, namely $2^{\Theta(n/\log n)}$, and our displayed tensor bound would not improve on it. This is an illustration of the comparison criterion, not an exhibited BKZ attack for the fixed-q, fixed-noise family. We must not use this hypothetical as evidence that the eventual advantage in that model fails.

If q, the noise width, or the sample budget instead vary with n, the original-modulus conventional cost must be reanalyzed for that family. For example the constant $C = C_0 q/\sqrt{c}$ in (6.11) ceases to be fixed when q grows, so the proof of Proposition 6.1 no longer establishes $\beta_q/d \to 1$. Independently, increasing W or the necessary m increases F. Subexponentiality of our own bound is therefore a distinct claim from asymptotic superiority over every original-instance baseline.

7.3 What the supplied source actually says about starting sizes

The ECDLP manuscript [3, Section 1.2] reports that savings from its assumed contraction algorithm require networks of size at least $2^{2^{17}}$. The source of this scale is visible in [1, Appendix D.4.2]: with

$$\lambda = \lceil \log_2 s \rceil, \qquad \epsilon = \frac{1}{128 \cdot 1024} = 2^{-17},$$

the root valuation parameter is $u_0 = \lfloor \epsilon \lambda \rfloor$. Below roughly $\lambda = 2^{17}$ it is zero, so that recursion provides no such saving. Rounding a logarithm can shift the edge threshold by a factor of two, irrelevant to its magnitude but not to its interpretation.

It is essential that this is a *necessary starting scale*, not a sufficient proof of speedup at that scale. The same appendix states several additional inequalities only for sufficiently large s. For example its display (D.5) uses $\lambda^{4/5} \leq u$. At the root, combining this particular sufficient condition with $u_0 \leq 2^{-17}\lambda$ already requires

$$\lambda^{1/5} \geq 2^{17}, \qquad \lambda \geq 2^{85}.$$

This is an audit of one of the proof's displayed sufficient inequalities, not a lower bound on every possible implementation or a claim that the algorithm cannot be reanalyzed at smaller parameters. Other recursion-depth conditions and constants must also be checked. It does show why the smaller $2^{2^{17}}$ quotation cannot be used as a complete numerical crossover theorem.

The independent SAT manuscript [2] also explicitly describes its bounds as purely asymptotic and gives enormous starting-size estimates. Its SAT thresholds concern its own inputs and proof, not our ring network; they cannot simply be substituted for S_0 in (7.1).

7.4 Concrete sizes and honest conclusions

For a 20-bit strict absolute error bound and $q = 2^{64}$, the retained modulus is $r = 23$. Even ignoring all constants, $nr/\sqrt{\log(nr)}$ does not have a small coefficient of n at ordinary cryptographic sizes. The former optimistic illustration $KA = 1$, $m = \rho = 1$ asks for $\log_2(nr)$ on the order of $6.2 \cdot 10^3$. To expose the encoding constant instead, the following are evaluations of the right side of (7.5) with $K = 1$, $m = 1$, $r = 23$, $\rho = 1$:

A	Certificate used	Required logarithmic scale
25	Compact interval network in the narrower range	$3.86 \cdot 10^6$
33	Compact interval network over the full stated range	$6.73 \cdot 10^6$
1000	Conservative reference construction	$6.18 \cdot 10^9$

The left side is $\log_2(Anr)$, rather than $\log_2(nr)$; that fixed additive shift is negligible at these scales. Neither $K = 1$ nor $\rho = 1$ is supplied by the source theorem or an estimator run for the intended instance. These numbers are bookkeeping illustrations, not numerical security estimates or predicted crossovers.

The source's nonzero-root scale is $\log_2 s$ of order 131072. The proof-condition audit above also gives $\log_2 s \geq 2^{85}$, of order $3.9 \cdot 10^{25}$, for that particular sufficient inequality. The three illustrative logarithmic scales in the table are larger than 131072 but vastly smaller than 2^{85}. Thus that additional source condition, not the encoding constant alone, still dominates this audit. It is not an inherent lower bound on every implementation. A concrete attack comparison also requires the original-modulus conventional estimate, its schedule, all constants, and a justified applicability range.

No conclusion about 128-bit, 192-bit, or 256-bit concrete security follows from these calculations. Reporting $r = 23$ as though it were a security exponent would confuse word width with the number of guessed bits. Similarly, polynomial space does not mean that the time bound is practical.

Padding a small network with forced-zero dummy edges preserves its contraction but makes the solver work at the padded size. It does not move a small cryptographic instance into a useful fast range for free. At small parameters the combined attack should simply choose the best available method, including binary-secret exhaustive search, conventional BKZ attacks, and any distribution-specific alternatives.

There is therefore a conditional *eventual* improvement over a stated $2^{\Theta(n)}$ BKZ/sieving benchmark whenever our subexponential regime holds. The fixed-modulus primal calculation supplies one explicit model with this behavior. There is no proved general BKZ lower bound, no universal superiority claim over other attack families, and no justified finite numerical crossover from the supplied asymptotic theorems alone.

8 Generalization 1: moduli that are not powers of two

The power-of-two hypothesis has two distinct uses: an exact small divisor, and the chain-ring collision argument. Neither means every other modulus is resistant to this approach.

8.1 A sufficiently large two-part already suffices

If $q = 2^t q_{\text{odd}}$ and contains the desired divisor $D = 2^r$, the binary attack is unchanged. Reduction maps uniform $a \in R_q$ to uniform $a \bmod D$. The recovered binary vector is the full secret, not only its two-primary component. Thus odd factors of the original modulus need not be attacked at all. The only extra cost is reading the original input. This assertion follows immediately from Lemma 2.5 and Theorem 5.1.

8.2 An odd prime divisor: a different image argument

Let p be an odd prime divisor of q, with $p > \max\{W, 2\}$, and work modulo p. Define $d_p = \text{ord}_{2n}(p)$ and $g_p = n/d_p$. The polynomial $X^n + 1$ has n distinct roots in an algebraic closure: its derivative is nX^{n-1}, coprime to it in odd characteristic. These roots have order $2n$, since n is a power of two. Frobenius sends a root ζ to ζ^p. Its orbit length is the smallest d with $p^d = 1 \pmod{2n}$, so each irreducible factor has degree d_p. The Chinese remainder theorem therefore gives

$$R_p \cong \prod_{i=1}^{g_p} \mathbb{F}_{p^{d_p}}. \tag{8.1}$$

Its ideals are specified by choosing which field components may be nonzero. An ideal supported on h components has dimension hd_p over $\mathbb{F}_p$.

Lemma 8.1 (A box in a vector space)**.** *If $V \subseteq \mathbb{F}_p^n$ has dimension d, and each coordinate is restricted to a set of at most K elements, the intersection contains at most K^d vectors. The same is true for a translate of V.*

Proof. A rank-d matrix whose columns form a basis has a nonsingular d-row minor. Projection onto those d coordinate positions is injective on V. There are at most K^d possible projected tuples. Translation preserves injectivity. □

Proposition 8.2 (Binary recovery through an odd factor)**.** *For m ring samples reduced modulo p, the probability of a competing binary secret is at most*

$$\left(1 + [2(W/p)^m]^{d_p}\right)^{g_p} - 1. \tag{8.2}$$

Proof. Stratify nonzero differences $s' - s$ by their exact nonzero field components in (8.1). A chosen support of size h gives an image ideal V of dimension $d = hd_p$. There are at most 2^d possible bounded differences in this ideal by Lemma 8.1, applied to the translated binary box. For each fixed difference, $a\delta$ is uniform on V. Its intersection with the translated interval box has at most W^d elements, so acceptance probability is at most $(W/p)^d$ per sample, even for fixed true errors. The m independent multipliers give $(W/p)^{md}$. Sum $[2(W/p)^m]^{hd_p}$ over all nonempty supports; the binomial theorem yields (8.2). □

There is no restriction to primes congruent to three or five modulo eight. For a fixed odd p, g_p is bounded as power-of-two n grows, while $d_p = \Theta(n)$. One can see this directly from the orders modulo powers of two. For $p \equiv 1 \pmod 4$, writing $a = v_2(p-1)$ gives $v_2(p^{2^j} - 1) = a + j$; each squaring multiplies by an odd factor after the first valuation

is extracted. For $p \equiv 3 \pmod 4$, apply the same argument to p^2, with $v_2(p^2-1)$ fixed. Thus the order eventually doubles whenever the required modulus doubles, and n/d_p stays bounded. In particular a fixed factor with $2(W/p)^m < 1$ gives exponentially small ambiguity.

For completeness, the odd-coefficient encoding can be made explicit. Let J be the least power of two with $J^2 \geq 4n$, and put $H = J/2$, $B = \lceil n/H \rceil$, $A = (\mathbb{Z}/p\mathbb{Z})[Y]/(Y^J+1)$. Pack blocks of length H. Then $B \leq J/2$, and the least power of two $L \geq 2B-1$ satisfies $L \leq J$, so $L \mid 2J$. Put $\omega = Y^{2J/L}$. For $0 < h < L$, write $h = 2^a h_0$ with h_0 odd. In the geometric sum, group each cycle's exponents in pairs separated by half a cycle. Their ratio is $Y^{Jh_0} = -1$, so each pair cancels. Thus the root is principal; L is a unit since p is odd. Fourier inversion follows by precisely the double-sum proof of Lemma 4.1.

The block index does not wrap because $L \geq 2B-1$. Each block product has degree at most $2H-2 < J$, so the auxiliary polynomial modulus introduces no aliasing. The factorization (4.3)–(4.4) applies with balanced power-of-two factors. Its six arrays contain $O(LJ\lceil \log p \rceil) = O(n \log p)$ bits and its degree is $O(n^{3/4} \log p)$. Represent every coefficient by its canonical integer in $[0, p-1]$. Each producing predicate requires this canonical range for its outputs, and input predicates require it for inputs; bit strings representing integers $\geq p$ are rejected. Hence unique extension holds. Recombination and negacyclic folding are unchanged. All predicates use polynomial-time modular arithmetic. This proves the claimed odd-modulus encoding, and applies unchanged to odd composite D with $\log D$ in place of $\log p$. It establishes the same subexponential regime when a suitable small odd divisor supplies the recovery bound.

8.3 Combining several small odd factors

We include a deliberately conservative bound that avoids requiring one factor to be larger than the noise. Let $D \mid q$ be odd and squarefree, with $D > 2W$. Define

$$a_s = \frac{\log 4}{\log D}, \qquad a_e = \frac{\log(2W)}{\log D}, \qquad \theta = m(1-a_e) - a_s.$$

Lemma 8.3 (Box counting in a residue submodule)**.** *For any additive submodule $V \subseteq (\mathbb{Z}/D\mathbb{Z})^n$ and a translated coordinate box of $K < D/2$ consecutive integers,*

$$|V \cap box| \leq |V|^{\log(2K)/\log D}.$$

Proof. Lift V to the full-rank lattice Λ of its integer representatives; then $D\mathbb{Z}^n \subseteq \Lambda \subseteq \mathbb{Z}^n$. A triangular Hermite basis has positive diagonal entries $h_i \mid D$. The divisibility follows by expressing De_i in the triangular basis, solving successively from its last nonzero coordinate. Its index is $\prod_i h_i$, hence $|V| = D^n/\prod_i h_i$. Successive solution of the triangular congruences restricts coordinate i to a class modulo h_i, with at most $\lceil K/h_i \rceil$ choices in an interval of K integers. Put $a = \log(2K)/\log D < 1$. If $h_i \geq K$ this count is at most one. Otherwise

$$\lceil K/h_i \rceil \leq 2K/h_i \leq (D/h_i)^a,$$

where the second inequality uses $a \leq 1$ and $D^a = 2K$. Multiplying gives $\prod_i (D/h_i)^a = |V|^a$. The box is injective modulo D, so this counts its residue intersection as well. □

Proposition 8.4 (A coupled CRT sufficient bound)**.** *If $\theta > 0$, the ambiguity probability for binary secrets is at most*

$$\prod_{p \mid D} (1 + p^{-\theta d_p})^{g_p} - 1. \tag{8.3}$$

Proof. For each possible nonzero multiplication ideal $V = \delta R_D$, the number of binary differences it contains is at most $|V|^{a_s}$ by Lemma 8.3. Each sample accepts a fixed

difference with probability at most $|V|^{a_e}/|V|$. Across m samples the total contribution for that ideal is at most $|V|^{-\theta}$. CRT classifies its support independently at every prime as in (8.1). Summing the products $\prod_p p^{-\theta d_p h_p}$ over all supports, excluding the zero ideal, proves (8.3). □

For example $D \geq (8W)^2$ makes $\theta \geq 1/2$ for one sample. If the selected factors and W are polylogarithmic in n, one can keep $\log D = O(\log \log n)$. Also $g_p \leq p + 1$ for powers-of-two n by the valuation calculation above, so $d_p \geq n/(p + 1)$, and (8.3) tends to zero rapidly. This gives another positive subexponential regime. It is sufficient, not optimized.

8.4 Where the non-power-of-two case remains difficult

If all usable factors have $\Theta(\log n)$ bits, the straightforward encoding has $O(n \log n)$ edges and only gives $2^{O(n\sqrt{\log n})}$. NTT-friendly primes $p \equiv 1 \pmod{2n}$ satisfy $p \geq 2n + 1$ and fall into this category as an asymptotic family. Small-factor projection is unavailable there. Moreover, projection to a factor of the *polynomial* is not the same as coefficientwise modulus reduction: it mixes error coordinates and need not preserve a small interval distribution. These are barriers to this particular proof, not hardness theorems for all other moduli.

9 Generalization 2: other defining polynomials

Let $f \in \mathbb{Z}[X]$ be monic of degree n, and consider $T_f = (\mathbb{Z}/2^r\mathbb{Z})[X]/(f)$. There are two separate questions: whether multiplication can be checked compactly, and whether bounded candidates have sufficiently few collisions. The first extends further than the second.

9.1 Compact multiplication extends to every public monic polynomial

An ordinary product of degree-below-n inputs has degree at most $2n - 2$. The convolution construction in Section 4 computes that ordinary product before its final negacyclic fold. For a general monic f, one can replace the fold by a constant number of further public polynomial multiplications. The following explicit division identity proves this.

For a polynomial H of degree at most d, define $\mathrm{rev}_d(H) = X^d H(X^{-1})$, with zeros padded to length $d + 1$. Let $P = Qf + R$, $\deg P \leq 2n - 1$, $\deg Q < n$, and $\deg R < n$. Reversing at the indicated fixed lengths gives

$$\mathrm{rev}_{2n-1}(P) = \mathrm{rev}_{n-1}(Q)\,\mathrm{rev}_n(f) + X^n\,\mathrm{rev}_{n-1}(R). \tag{9.1}$$

The constant coefficient of $\mathrm{rev}_n(f)$ is one. It therefore has a unique inverse h modulo X^n over *any* coefficient ring. Indeed if $g_0 = 1$, coefficients of h are determined by $h_0 = 1$ and $h_j = -\sum_{i=1}^{j} g_i h_{j-i}$. They are public and computable in polynomial time.

Reducing (9.1) modulo X^n yields

$$\mathrm{rev}_{n-1}(Q) = \mathrm{rev}_{2n-1}(P)h \pmod{X^n}.$$

One public-times-variable convolution determines this quotient. A second convolution computes Qf, and coefficientwise subtraction gives R; the high coefficients can also be checked to vanish. Together with the initial product this is a constant number of linear-size gadgets. Reversal, truncation, and subtraction add only $O(nr)$ edges and low degree. Every intermediate is unique. Hence public multiplication modulo an arbitrary monic

f has an $O(nr)$-edge encoding with degree $O(n^{3/4}r)$, up to different absolute constants. Public coefficients are first reduced modulo 2^r; a dense public f is not an exponential table.

For sparse polynomials the final step can be even simpler. Modulo $X^n - X - 1$, for example, $X^{n+j} = X^{j+1} + X^j$ for $j \le n-2$. Each output receives at most three ordinary coefficients, so one local linear layer suffices. We do not assert this polynomial is irreducible modulo two for arbitrary n.

9.2 A positive case: irreducibility modulo two

Proposition 9.1 (Other rings can have an easier recovery proof)**.** *Suppose $f \bmod 2$ is irreducible of degree n. A nonzero difference of two binary coefficient vectors is a unit in T_f. Consequently, for uniform public multipliers and interval errors of width $W < D$, the ambiguity probability from m samples is at most*

$$[2(W/D)^m]^n.$$

With $q \ge 4W$, this yields the same subexponential regime as the main theorem, with no parity correction.

Proof. A nonzero binary difference remains a nonzero degree-below-n polynomial modulo two. Irreducibility makes it invertible in $\mathbb{F}_2[X]/(f)$. An inverse lifts to modulo 2^r by the finite geometric series or Newton argument from Lemma 2.4. Multiplication by the difference is therefore a permutation of T_f, so $a\delta$ is uniform on the entire coefficient space. For any fixed error vector, exactly W^n of the D^n possible images produce another allowed residual. Multiply over samples and union-bound over competing binary secrets. The compact encoding was proved above. □

This positive case shows that the particular polynomial $X^n + 1$ is not essential to the tensor technique. Its special role in the main theorem is to give a provable image-size statement despite ramification at two.

9.3 A broader sufficient condition for squarefree reductions

Suppose f mod 2 is squarefree, with g irreducible factors of degrees $d_1, \dots, d_g$, and $d_{\min} = \min_i d_i$. Here is a proof of the structural facts needed for a sufficient bound. Pairwise coprime monic factors lift from modulo 2^h to modulo 2^{h+1}: if $f - gh = 2^h E$, solve $h\Delta g + g\Delta h = E \pmod 2$ using a Bezout identity between g, h, then replace the factors by $g + 2^h\Delta g$ and $h + 2^h\Delta h$, reducing degrees by polynomial division. Iterate and apply this to all factors. The lifted factors remain comaximal because a Bezout identity equal to one modulo two lifts to a unit identity. CRT therefore expresses T_f as a product of rings of respective sizes D^{d_i} with field reductions modulo two.

In each component, an element not divisible by two is a unit by inverse lifting; otherwise extract the greatest common coefficient power of two. Thus its ideals are (2^v), $0 \le v \le r$. There are at most $(r+1)^g$ ideals in the product. A nonzero binary difference is nonzero modulo two in at least one component, where it is a unit. Its multiplication image has size at least $D^{d_{\min}}$.

Use the box exponents from Section 8, now over $\mathbb{Z}/D\mathbb{Z}$: $a_s = \log 4/\log D$, $a_e = \log(2W)/\log D$, and $\theta = m(1-a_e) - a_s > 0$. The proof of Lemma 8.3 did not require a squarefree integer modulus. Stratifying by multiplication ideals exactly as before gives

$$\Pr[\text{ambiguity}] \le (r+1)^g D^{-\theta d_{\min}}. \tag{9.2}$$

For fixed r, g, $d_{\min} = \Omega(n)$, and fixed noise width this is exponentially small. More factors or small-degree components weaken this particular bound. No generic statement about all other number fields follows.

9.4 A concrete obstruction to the original chain-ring proof

For the cyclic quotient $f = X^n - 1$, let $\delta = 1 + X + \cdots + X^{n-1}$. Then

$$a\delta = (a_0 + \cdots + a_{n-1})\delta \pmod{X^n - 1}.$$

Its multiplication image has only D elements, so for $n > 1$ it need not contain the $2^{n(r-1)}$ elements of $2T_f$. This is a small-coefficient difference for which the main image lemma fails. It does not prove the cyclic problem hard or rule out a distribution-specific average-case argument. It proves that the earlier proof cannot be transplanted unchanged.

Finally, changing the defining ring may change what "small error" means. An error spherical in a canonical embedding need not have independent bounded coefficients in the chosen power basis. The statements here concern the explicitly specified coefficient distribution. A change-of-basis distortion and its correlations must be analyzed before importing them into another Ring-LWE formulation.

10 Generalization 3: secrets that are not binary

10.1 Bounded alphabets

Suppose every secret coefficient lies in a known interval of L integers. Let $H = 2^{\lceil \log_2 L \rceil}$ and take $D > H$. A nonzero difference has a coefficient of nonzero absolute value below H, so its multiplication ideal contains HT by Corollary 2.2. There are at most L^n candidate secrets. Lemma 3.1 gives, for any independent bounded noise law and m uniform multipliers,

$$\Pr[\text{ambiguity}] \le \left[L \left(\frac{H \lceil W/H \rceil}{D} \right)^m \right]^n. \tag{10.1}$$

This proves the extension without requiring a uniform secret distribution. If $D \ge L$, a secret coefficient has a unique lift to its interval. Interval membership on the input words adds only $O(nr)$ constraints.

For ternary secrets $L = 3$, $H = 4$. With ternary errors $W = 3$, one sample modulo $D = 32$ has ambiguity at most $(12/32)^n = (3/8)^n$. For binary secrets and the same errors, $D = 16$ gives at most 2^{-n}. These are exact distributional recovery bounds, not concrete runtime claims.

One sample and a modulus of size $D = O(L(W + L))$ suffice with a fixed reliability margin, since $H \lceil W/H \rceil < W + H$ and $H < 2L$. The retained width is therefore $r = O(\log(L + 1) + \log(W + 1))$. Fixed or sufficiently slowly growing alphabets preserve the subexponential regime; arbitrary growing alphabets do not automatically do so. For a fixed-weight or otherwise restricted secret set of size N_s, replace L^n in the union bound by N_s and add the restriction to the network. One must then compare against direct enumeration of that smaller set.

10.2 Unrestricted secrets can instead be eliminated

A uniform secret in R_q has nt bits, but that observation alone does not rule out our approach. With bounded errors one can cancel the secret and use errors as the small witness. This is an important positive result, not a claimed impossibility for nonbinary secrets.

Let the errors lie in the interval I of width W. Choose

$$H_W = 2^{\lceil \log_2 W \rceil}, \qquad D = 2H_W^2 \mid q.$$

Take two ring samples $(a_0, b_0), (a_1, b_1)$, with a_0 a unit. This is a publicly testable event of probability $1/2$; more samples can supply a unit base. By Lemma 2.4, being a unit modulo D already means being a unit modulo the original power-of-two q. Define modulo D

$$A = a_1 a_0^{-1}, \qquad c = Ab_0 - b_1.$$

Then $c = Ae_0 - e_1$. Conditional on the base being a unit, A is uniform and independent of both errors. Search for $(e_0', e_1') \in I^n \times I^n$ satisfying that relation.

Proposition 10.1 (Error-first recovery for any secret)**.** *Conditional on a unit base, the error pair is unique except with probability at most* $(WH_W/D)^n \le 2^{-n}$*. When unique, it determines the original secret, regardless of its distribution. The conditional tensor complexity uses* $r = 1 + 2\lceil \log W \rceil$ *bits and remains* $2^{O(nr/\sqrt{\log(nr)})}$*, plus polynomial input work.*

Proof. A false e_0' gives a nonzero difference with coefficients of absolute value at most $W - 1 < H_W$. Its image contains $H_W T$. In each scalar class modulo H_W, the translated error interval has at most one element, since $W \le H_W$. Thus a fixed false e_0' yields an allowed e_1' with probability at most $(H_W/D)^n$. There are at most W^n choices, proving the bound. Here $WH_W/D = W/(2H_W) \le 1/2$.

The true pair is always feasible. Since $W < D$, residues of the recovered errors lift uniquely to the original integer interval. Now compute $s = a_0^{-1}(b_0 - e_0) \pmod q$ using the original sample and original modulus. This is exact. The network of Section 4 checks $Ae_0' - e_1' = c$, with the input and output interval predicates replacing the binary restriction. All its auxiliary variables remain unique. Counting and bitwise self-reduction now require at most $nr + 1$ calls, because the witness is an nr-bit error vector rather than an n-bit binary secret; this is a polynomial factor. □

If a fresh unit base is sampled until one is found, the number of samples needed for the base is geometric with expectation two; one further sample supplies the second equation. The input generation must permit this, or the analysis must include the probability that the available list has no unit. The algorithm does not manufacture independent samples from rotations.

The one-relation modulus scale also has a direct information-theoretic explanation for uniform independent errors. Conditional on any fixed A, the W^{2n} pairs (e_0, e_1) are equally likely and the label $c = Ae_0 - e_1 \pmod D$ has at most D^n values. A decoder choosing one pair per label has success probability at most D^n/W^{2n}. When $W < D$, recovering e_0 determines e_1 uniquely from the label and its interval, so the same bound applies to that recovery task. Thus reliable recovery of the pair from this relation alone needs D at least of order W^2. This does not rule out different representations, additional relations, nonuniform error distributions, or use of a restrictive secret promise. It is a lower bound for this specified uniform-pair task, not an assertion that the earlier collision upper bound is tight in every detail.

For j normalized relations with one shared base error, the same proof gives

$$\Pr[\text{ambiguity}] \le \left[W(H_W/D)^j\right]^n.$$

This trades additional samples against a smaller available divisor. Its encoding has $O(jnr)$ edges. For one relation the retained modulus is quadratic in W, rather than linear as with a bounded secret. Nevertheless the condition $\log W = o(\sqrt{\log n})$ is unchanged in order of growth.

For the $q = 2^{64}$ example with $|e_i| < 2^B$, binary recovery uses $r = B + 3$. The two-sample error-first choice uses $r = 2B + 3$. At $B = 20$ these are 23 and 43 bits, both available; at larger widths the latter may require more normalized relations. The difference is quantitative, not a blanket loss of subexponentiality for unrestricted secrets.

If the noise is unbounded, a small-modulus error residue need not lift to the actual integer error, so this last recovery step is no longer automatic. That is a genuine distinction between an unrestricted secret and a binary secret, which itself lifts uniquely. Section 12 makes the corresponding Gaussian extension precise for small secrets.

11 Generalizations 4 and 5: dense LWE and lattice reduction

11.1 Why the direct dense-LWE encoding is too large

For ordinary binary-secret LWE, a scalar sample is $b_j = \sum_{i=1}^{n} a_{j,i}s_i + e_j \pmod q$. There is no shared convolution among independently chosen dense rows. Even an optimistic encoding can put one entire scalar equation in a predicate reading n binary secret bits; the large public coefficients and interval test are evaluated internally. Consistency trees connect the copies across m equations. This gives $O(mn)$ Boolean edges, without even charging r bits per binary coefficient. For $m = \Theta(n)$ the assumed solver only yields

$$2^{O(n^2/\sqrt{\log n})},$$

which does not improve on 2^n enumeration. A standard arithmetic circuit can introduce additional word-width costs, not remove this basic obstacle.

Why not put the entire verification in one predicate? Its n input ports would violate the degree cap for a network of $O(n)$ edges. One could pad the graph to $\Theta(n^{10/9})$ edges to make the degree legal, but then the bound becomes $2^{O(n^{10/9}/\sqrt{\log n})}$, still worse than exhaustive search. This argument shows a failure of these encodings, not a lower bound against every encoding or every algorithm.

For a uniform secret, selecting a square invertible block of sample rows and changing variables to its errors produces the familiar normal-form idea. However its inverse times the remaining rows is generally dense. It does not create the public convolution relation of Proposition 10.1. Keeping the inverse implicit by reintroducing all original equations merely returns to the dense verifier. No compact linear-size construction for arbitrary dense matrices has been proved here.

11.2 Sparse public matrices are a genuine special case

For completeness, suppose $A \in \mathbb{Z}_q^{n\times m}$ has at most k nonzero entries per column and the secret is uniform. This is the sparse-matrix LWE model considered in [11]; it is different from a sparse secret. After reduction to $D = 2^r \mid q$, one sample predicate reads only kr secret bits. Equality trees and canonical range checks give

$$S = O((n + mk)r).$$

On independent uniform labels, a fixed candidate secret satisfies all interval constraints with probability $(W/D)^m$. A union bound over D^n secrets gives acceptance probability at most $D^n(W/D)^m$. On planted bounded-noise samples, a satisfying secret always exists. Thus $D \geq W^2$, $m = 3n$ give a conditional distinguisher with false acceptance at most W^{-n} and time

$$2^{O(nkr/\sqrt{\log(nkr)})}.$$

The proof requires neither uniqueness nor matrix rank. It breaks a decisional assumption in suitable small-k, small-r regimes. At logarithmic k, even constant r gives only $2^{O(n\sqrt{\log n})}$ from this particular bound. Sparse matrices therefore supply additional positive cases but do not turn the dense-LWE claim into a theorem.

11.3 Why this is not a generic SVP algorithm

A general SVP verifier takes an integer basis B and a coefficient vector z, checks $v = Bz \neq 0$, and checks $\|v\| \leq R$. Three obstacles replace the promises used in the ring attack:

1. The map $z \mapsto Bz$ is a general dense matrix map, not multiplication by one public ring element.
2. Short v need not have binary or small coefficients z in the supplied basis.
3. The required coefficient precision and the metric of a projected block depend on the input basis and the projection. A verifier polynomial in input length is not necessarily linear in block dimension.

Here is a quantitative illustration of the second issue. Suppose a basis of dimension d has size-reduced coefficients $|\mu_{j,i}| \leq 1/2$ and the usual LLL inequalities with parameter $3/4$. The latter imply $\|b_i^*\| \geq 2^{-(i-1)/2} \|b_1\|$, because each adjacent ratio has square at least $3/4 - 1/4 = 1/2$. Any vector $v = \sum_i z_i b_i$ with $\|v\| \leq \|b_1\|$ satisfies

$$\left| z_i + \sum_{j>i} \mu_{j,i} z_j \right| \leq \frac{\|v\|}{\|b_i^*\|} \leq M := 2^{(d-1)/2}.$$

Back substitution gives

$$|z_i| \leq M(3/2)^{d-i}. \tag{11.1}$$

To prove the induction, bound the tail by $M \sum_{h=0}^{d-i-1} (3/2)^h = 2M((3/2)^{d-i} - 1)$ and use the factor $1/2$ in the size-reduction coefficients. Thus a standard complete coefficient certificate can require $O(d)$ bits per coefficient and $O(d^2)$ bits overall. This is an upper-bound construction, not a proof that a shorter certificate is impossible. It does explain why applying a subexponential algorithm in the certificate length does not establish subexponential time in d.

The same issue recurs when attempting to replace BKZ's SVP oracle. After an arbitrary unimodular change and orthogonal projection, a short target need not be short in coefficient space. The algorithm in this paper has not been proved correct for those oracle inputs. Its interval predicates would reject legitimate SVP witnesses unless new bounds and encodings were supplied.

11.4 Approximation, reduction direction, and remaining work

Solving a particular average-case Ring-LWE distribution does not by itself give an arbitrary-lattice reduction algorithm. Known worst-case/average-case connections have a specified direction, approximation factor, field class, noise distribution, modulus conditions, and often dimension or module-rank losses. One cannot discard those parameters or assume that a theorem for Gaussian errors automatically covers this exact bounded uniform-error model. Nor does refuting ETH imply polynomial time for all NP problems.

A generic basis-reduction improvement would require a separate theorem: given a projected rank-β lattice and a target radius or approximation factor, construct a permitted network of size $o(\beta\sqrt{\log \beta})$, with controlled bit precision, and recover the required vector. The inequality follows directly by demanding that $s/\sqrt{\log s} = o(\beta)$. No such general theorem has been proved in this manuscript. Structured subfamilies with sparse checks or compact algebraic maps remain plausible targets, but they are separate research problems rather than consequences of the Ring-LWE proof.

12 Uniform versus Gaussian-like noise

The support width alone does not describe a noise distribution. Concentrating probability near zero can help recovery, but the counting solver does not automatically explore likely witnesses first. This section proves how to exploit concentration without losing the linear-size encoding. It also sharpens the collision argument for binary secrets: a valuation-stratified bound removes one bit of overhead from a generic entropy-based criterion. All assertions concern independent errors, independent of the uniform public multiplier. Correlated errors require separate analysis.

12.1 Same support, different probabilities

Fix an integer $C \geq 1$ and the common support

$$I = \{-C, \dots, C\}, \qquad W = 2C + 1.$$

Compare uniform interval noise with the truncated discrete Gaussian

$$P_\lambda(z) = \frac{e^{-\lambda z^2}}{Z_C(\lambda)}, \qquad Z_C(\lambda) = \sum_{j=-C}^{C} e^{-\lambda j^2}, \qquad \lambda = \frac{1}{2\sigma_0^2} > 0. \tag{12.1}$$

The shape parameter σ_0 is not quite the actual standard deviation after truncation. Write $v = \mathbb{E}E^2$ and $\sigma = \sqrt{v}$ for the actual variance and standard deviation; symmetry gives mean zero. Uniform noise has variance $C(C+1)/3$ and entropy $\log_2 W$.

If the verifier still checks only that every residual lies in I, its graph and running-time bound do not change. The support-only recovery bound of Theorem 3.3 remains valid for every noise law on I, not only the uniform law. Thus changing probabilities does not force a worse support-based guarantee. The special uniform-noise parity bound need not hold verbatim for the new law, but the one-sample choice of Corollary 3.4 is unchanged. Any further benefit requires an explicitly different acceptance test, such as a typical-set or norm test.

Concentration reduces both variance and entropy in (12.1). Differentiating finite sums gives

$$v = -\frac{d}{d\lambda} \ln Z_C(\lambda), \qquad H(P_\lambda) = \frac{\ln Z_C(\lambda) + \lambda v}{\ln 2}. \tag{12.2}$$

Moreover $v' = -\operatorname{Var}(E^2) \leq 0$, so

$$\frac{dH(P_\lambda)}{d\lambda} = -\frac{\lambda}{\ln 2} \operatorname{Var}(E^2) \leq 0.$$

Uniform noise is the limit $\lambda \to 0$. More generally, entropy on W points is at most $\log_2 W$: Jensen's inequality gives $\sum_z p_z \log(1/p_z) \leq \log(\sum_z p_z/p_z) = \log W$ on full support.

12.2 Why discarding each coefficient's tails can fail

Choose a smaller fixed coordinate cutoff $c < C$ and let $p = \Pr[|E| > c] > 0$. Independence gives

$$\Pr[|E_i| \leq c \text{ for every } i] = (1-p)^n \longrightarrow 0.$$

Thus even a tiny fixed per-coordinate tail probability eventually rejects almost every true error vector. A Gaussian coordinatewise cutoff usually has to grow with n to avoid this problem. Merely replacing C by a few standard deviations in the original interval verifier is not an asymptotically valid argument.

Instead accept a high-probability *set of vectors*. A total squared norm test permits occasional tail coefficients while controlling their aggregate contribution. Its true-acceptance probability and the number of false explanations can be analyzed separately.

12.3 A sharper binary-secret collision bound for any acceptance set

Let $D = 2^r \mid q$ and identify R_D with coefficient residue vectors. No relation between D and W is required here: true errors may wrap modulo D. Binary secrets still lift uniquely to the original modulus. Write $Y = X + 1$, as in Lemma 2.1.

The exact valuation counts needed here were proved in Lemma 3.5. They do not depend on the noise distribution.

Theorem 12.1 (Acceptance-set recovery bound)**.** *Let $\mathcal{T} \subseteq R_D$ be any deterministic accepted set of error residues. For one sample, every fixed binary secret and fixed true error vector satisfy*

$$\Pr_a[\exists s' \neq s : \ b - as' \in \mathcal{T}] \leq \frac{n}{2} \frac{2^n|\mathcal{T}|}{D^n}. \tag{12.3}$$

If $\Pr[e \bmod D \notin \mathcal{T}] \leq \varepsilon_n$, counting and recovering the unique accepted binary secret is correct except with probability at most

$$\varepsilon_n + \frac{n}{2} \frac{2^n|\mathcal{T}|}{D^n}.$$

Proof. For a fixed competitor of valuation j, $a(s' - s)$ is uniform on an ideal of size $D^n/2^j$. Its intersection with $e - \mathcal{T}$ has at most $|\mathcal{T}|$ elements. Acceptance has probability at most $|\mathcal{T}|2^j/D^n$. Union-bound using Lemma 3.5:

$$\sum_{j=0}^{n-1} 2^{n-j-1} \frac{|\mathcal{T}|2^j}{D^n} = \frac{n}{2} \frac{2^n|\mathcal{T}|}{D^n}.$$

This is uniform in the true error, so it can be averaged over its law. Unless the true residue is rejected or a false candidate is accepted, there is exactly one accepted secret and it is the true one. □

If $|\mathcal{T}| \leq A^n$, the false-candidate bound is $(n/2)(2A/D)^n$. The size-based sufficient threshold is therefore $D > 2A$ with fixed relative slack. This improves on using only the worst image size $(D/2)^n$, which would give $(4A/D)^n$. It does not uniformly dominate the sharper product-interval bounds in Section 3; those can avoid the factor $n/2$. The new argument exploits the exact binary alphabet and is not asserted for general interval secrets.

Unlike the full-support algorithm, a restricted acceptance set can reject the true secret. A unique accepted output is therefore not an unconditional correctness certificate under the original noise promise. The displayed sum bounds failure *or wrong output*; both terms must be retained. Checking the original-modulus residual may reject additional false outputs but does not by itself establish a zero-error guarantee.

Theorem 12.2 (Parity-sensitive acceptance-set bound)**.** *Suppose error coordinates are independent of one another and of the uniform public multiplier, and let*

$$b_i = \mathbb{E}(-1)^{E_i}, \qquad b = \max_i |b_i| \leq 1.$$

For a deterministic acceptance set $\mathcal{T} \subseteq R_D$ and every fixed binary secret,

$$\Pr_{a,e}[\exists s' \neq s : \ b_{\mathrm{pub}} - as' \in \mathcal{T}] \leq \frac{|\mathcal{T}|}{D^n} S_n(b) = R_n(b) \frac{2^n|\mathcal{T}|}{D^n}. \tag{12.4}$$

Here b_{pub} is the sample label, distinguished from the parity-bias parameter b. Always $R_n(b) \leq n/2$. If $b \leq b_0 < 1$ uniformly along a family, then $R_n(b) = 1 + o(1)$ by Lemma 3.7.

Proof. Fix a competitor of valuation $j < n$, with image $\Lambda_j = (Y^j)$. Since Λ_j contains $2R_D$, membership in it is equivalent to parity membership in $\mathcal{V}_j$. Its size is $D^n/2^j$, so

$$\Pr[e - a(s' - s) \in \mathcal{T}] = \frac{2^j}{D^n} \sum_{t \in \mathcal{T}} \Pr[\bar{e} \in \bar{t} + \mathcal{V}_j].$$

The subspace indicator has the character expansion

$$\mathbf{1}_{\bar{t}+\mathcal{V}_j}(\bar{e}) = 2^{-j} \sum_{u \in \mathcal{V}_j^\perp} (-1)^{\langle u, \bar{e} - \bar{t} \rangle}.$$

This follows by summing characters: the sum is 2^j on the subspace and zero outside it. Independence of error coordinates gives

$$\Pr[\bar{e} \in \bar{t} + \mathcal{V}_j] = 2^{-j} \sum_{u \in \mathcal{V}_{n-j}} (-1)^{\langle u, \bar{t} \rangle} \prod_{i \in \mathrm{supp}(u)} b_i \le 2^{-j} A_j(b).$$

The per-competitor bound is therefore $|\mathcal{T}| A_j(b)/D^n$. Summing the exact numbers of competitors from Lemma 3.5 proves (12.4). Monotonicity of S_n on $[0,1]$ and $S_n(1) = (n/2)2^n$ prove the universal comparison, and the uniform limit was proved earlier. □

This bound averages over the error law. The pointwise guarantee of Theorem 12.1 remains useful when the error vector is fixed, or independence and a parity-bias bound are unavailable. For balanced independent parities, $b = 0$ gives the exact factor $S_n(0) = 2^n - 1$. For uniform interval noise, $b = 0$ for even W and $b = 1/W$ for odd W. Applying (12.4) to its interval box uses $S_n(1/W)$; exploiting the box's own Fourier coefficients gives the stronger exact first moment with $S_n(1/W^2)$ in Theorem 3.6. These are different statements.

For any fixed nondegenerate parity law the relative factor tends to one, but it need not be at most one at finite n. Nor does $S_n(b) = 2^n(1 + o(1))$ assert the stronger additive statement $S_n(b) = 2^n - 1 + o(1)$. Near-degenerate families need care: if $b_n = 1 - 1/n^2$, positivity of coefficients gives $S_n(b_n) \ge b_n^n S_n(1)$, so $R_n(b_n)$ remains asymptotic to $n/2$, even though each $b_n < 1$. Use the finite factor or an explicit bound such as (3.6).

12.4 Entropy gives an effective width and a near-matching threshold

Let the errors be independent with a fixed integer law, and let P be the induced law modulo a fixed D. Define Shannon entropy in bits by $h = H(P) = -\sum_z P(z) \log_2 P(z)$, omitting zero terms. Let $b = |\mathbb{E}(-1)^E|$ be its parity bias. For every fixed $\epsilon > 0$ there are sets $\mathcal{T}_n$ with

$$|\mathcal{T}_n| \le 2^{n(h+\epsilon)}, \qquad \Pr[e \bmod D \notin \mathcal{T}_n] \le e^{-cn}, \tag{12.5}$$

for sufficiently large n and some $c > 0$.

Here is an explicit proof. Accept all empirical distributions within a sufficiently small fixed coordinatewise tolerance of P, excluding symbols of probability zero. Entropy is continuous on the finite simplex, so each accepted type has entropy at most $h + \epsilon/2$. There are at most $(n+1)^D$ types. A type Q has at most $2^{nH(Q)}$ strings: under product law Q every such string has probability $2^{-nH(Q)}$, and their probabilities sum to at most one. The polynomial type count is absorbed into $2^{n\epsilon/2}$.

For concentration, a Bernoulli variable's centered log moment-generating function has second derivative at most $1/4$, its variance under an exponential tilt. Its value and derivative at zero vanish, so its value at θ is at most $\theta^2/8$. Markov's inequality with $\theta = 4\delta$ bounds either frequency deviation by $e^{-2n\delta^2}$. Union-bound over the fixed D symbols to prove the second part of (12.5).

These sets have linear-size encodings for fixed D. A local predicate converts each r-bit residue to its D indicators. For each symbol, add the indicators with a balanced binary tree and compare with the allowed frequency interval. At level j there are $n/2^j$ sums with $O(j)$ bits, so $\sum_j (n/2^j)j = O(n)$. Constants may depend strongly on D. For a computable law, rational approximations to the frequencies with strict tolerance slack suffice; irrational probability constants need not be manipulated exactly by the verifier.

Theorem 12.2 now gives

$$\Pr[\text{incorrect typical explanation}] \le R_n(b)\, 2^{-n(r-h-1-\epsilon)}. \tag{12.6}$$

For a fixed law with $b < 1$, $R_n(b) = 1 + o(1)$; without that assumption $R_n(b) \le n/2$ still suffices. Thus $r > h + 1$ with fixed slack suffices for exponentially reliable recovery under the assumed solver. This improves the coarser sufficient condition $r > h + 2$ obtained using only the minimum image size. The claim concerns a fixed finite residual law; for growing alphabets the histogram constants and the type-count factor must not be hidden.

Proposition 12.3 (An entropy converse)**.** *For a uniform binary secret, one ring sample modulo D, and independent noise residues of per-coefficient entropy h, any decoder with error probability p_n satisfies*

$$r - h \ge 1 - p_n - h_2(p_n)/n,$$

where h_2 is binary entropy. In particular, success tending to one requires $r - h \ge 1 - o(1)$.

Proof. Let S, A, B denote secret, multiplier, and label. Conditional on A, S, the label is a translate of the error, so $H(B \mid A, S) = nh$. Also $H(B \mid A) \le nr$, hence mutual information obeys $I(S; B \mid A) \le n(r - h)$. If F indicates a decoding error, the chain rule gives $H(S \mid A, B) \le H(F \mid A, B) + H(S \mid F, A, B) \le h_2(p_n) + p_n n$: when decoding is correct the output determines S, and otherwise at most 2^n secrets remain. Since $H(S \mid A) = n$, subtraction proves the inequality. Random decoder coins may be included in the observation without affecting the information bound. □

The statistical threshold therefore matches up to strict slack in this fixed-law setting. It is not a practical running-time estimate. Moreover $H(E \bmod D) \le H(E)$ whenever integer entropy is finite, because $H(E) = H(E \bmod D) + H(E \mid E \bmod D)$. Thus $r > H(E) + 1$ is a convenient stronger condition even if errors wrap modulo D. Integer error lifts need not be recovered to identify a binary secret. For unrestricted secrets, the error-first lifting step remains a separate issue, as discussed in Section 10.

For an interval secret alphabet of size L, the general image argument still gives the sufficient condition $r > h + \log_2(LH)$, where $H = 2^{\lceil \log_2 L \rceil} < D$. The one-bit binary refinement does not automatically extend to that case.

12.5 A compact norm test and its full reliability bound

A histogram with D counters is unattractive when D has millions of values. A norm test supplies a uniformly compact substitute. Suppose independent identically distributed integer errors have second moment v and a finite moment-generating function for E^2 near zero. Choose a known rational $V > v$. Accept centered residue vectors $z \in [-D/2, D/2)^n$ satisfying

$$\|z\|^2 \le \lfloor nV \rfloor .$$

Centering modulo D never increases absolute value. For any $\theta > 0$ where the moment exists, Markov's inequality gives

$$\Pr[\text{true residue rejected}] \le \exp\{-n[\theta V - \ln \mathbb{E} e^{\theta E^2}]\}. \tag{12.7}$$

The derivative of the bracket at zero is $V - v > 0$, so some positive constant exponent is available for every fixed law and strict slack.

For bounded $|E| \le C$, an explicit alternative is

$$\Pr[\text{rejection}] \le \exp\left(-\frac{2n(V-v)^2}{C^4}\right),$$

when $V < C^2$; if $V \ge C^2$, rejection is impossible. To prove it, apply the preceding exponential-moment argument to $E^2 \in [0, C^2]$. Under an exponential tilt its variance is at most $C^4/4$, so the centered log moment-generating function is at most $\theta^2 C^4/8$; optimize $\theta = 4(V-v)/C^4$.

The number of integer vectors in the accepted ball is at most

$$A_V^n, \qquad A_V = \sqrt{2\pi e}(\sqrt{V} + 1/2). \tag{12.8}$$

Disjoint unit cubes centered at these vectors lie in the ball of radius $\sqrt{n}(\sqrt{V} + 1/2)$. For even n, unit-ball volume obeys $\omega_n \le (2\pi e/n)^{n/2}$, since $(n/2)! \ge (n/(2e))^{n/2}$, obtained by integrating $\ln x$. The enclosing volume proves (12.8).

Combining the bounds gives the explicit one-sample result

$$\Pr[\text{failure or wrong output}] \le e^{-nI(V)} + R_n(b)\left(\frac{2A_V}{D}\right)^n, \tag{12.9}$$

where $b = |\mathbb{E}(-1)^E|$ and $I(V) > 0$ can be any established Chernoff exponent from (12.7). A clipped bound of one is understood. For false-candidate exponent $\gamma > 0$, choosing $D \ge 2^{1+\gamma} A_V$ gives $R_n(b) 2^{-\gamma n}$ for that term. The factor is $1 + o(1)$ for fixed $b < 1$ and at most $n/2$ universally. For a finite guarantee of at most $2^{-\gamma n}$, it suffices to use $D \ge 2^{1+\gamma} A_V R_n(b)^{1/n}$, rounded up to an available divisor. To obtain the same overall exponent, also check $I(V) \ge \gamma \ln 2$ with appropriate slack. Choosing V extremely close to v weakens the rejection exponent.

For the exact truncated Gaussian the tradeoff has the finite-sum expression

$$I(V) = \sup_{\theta > 0}\{\theta V - \ln Z_C(\lambda - \theta) + \ln Z_C(\lambda)\}. \tag{12.10}$$

The sums are finite, so negative $\lambda - \theta$ is allowed. Substitute $\mathbb{E}e^{\theta E^2} = Z_C(\lambda - \theta)/Z_C(\lambda)$ in (12.7). Positive exponents follow already by the derivative argument, without numerical optimization.

For comparison, a continuous $N(0, \sigma^2)$ has $\mathbb{E}e^{\theta E^2} = (1 - 2\theta\sigma^2)^{-1/2}$ by Gaussian integration. At $V = u\sigma^2$, $u > 1$, optimizing gives $I(V) = (u - 1 - \ln u)/2$. This illustrates the tradeoff; the actual discrete truncated law uses (12.10) instead.

12.6 The norm test preserves the tensor size bound

A local predicate converts each r-bit residue to its centered integer square, represented in $2r$ bits. The conversion is deterministic and polynomial-time. Sum the squares with a balanced tree. At level j there are $n/2^j$ sums of width $O(2r + j)$, hence

$$\sum_{j \ge 1} (n/2^j)(2r + j) = O(nr).$$

The final comparison reads $O(r + \log n)$ bits. Canonical representations force all intermediate values uniquely, so the network still counts secrets rather than auxiliary assignments. The added maximum degree is $O(r + \log n)$, compatible with the convolution network's degree cap.

Assumption 1.1 therefore still gives

$$\mathrm{poly}(n,t) + 2^{O(nr/\sqrt{\log(nr)})} \tag{12.11}$$

and polynomial space. Constants change: a square-and-sum network costs more than independent interval checks. Saving one retained bit alone does not prove a smaller concrete solver runtime. The support-only algorithm remains an available alternative. The compact binary construction of Section 4.6 gives explicit certificates of $33nr$ in its full $n \geq 8$, $2 \leq r \leq n^{3/2}$ range, or $32nr$ under (4.9), after accounting for the norm-test edges.

12.7 How much concentration helps at the same hard bound

The entropy in (12.2) is given by finite sums. Its familiar Gaussian approximation follows directly as well. Put $\kappa = C/\sigma_0$, let ϕ and Φ be standard normal density and cumulative distribution, and write $p_\kappa = 2\Phi(\kappa) - 1$. As $\sigma_0 \to \infty$ at fixed κ (with integer endpoint rounding), Riemann sums and integration by parts give

$$Z_C \sim \sigma_0\sqrt{2\pi}\,p_\kappa, \qquad \frac{v}{\sigma_0^2} \longrightarrow 1 - \frac{2\kappa\phi(\kappa)}{p_\kappa}.$$

Indeed $\int_{-\kappa}^{\kappa} x^2\phi(x)\,dx = p_\kappa - 2\kappa\phi(\kappa)$ using $\phi'(x) = -x\phi(x)$. Substitution in (12.2) yields

$$H(E) = \log_2(\sqrt{2\pi e}\,\sigma_0) + \log_2 p_\kappa - \frac{\kappa\phi(\kappa)}{p_\kappa \ln 2} + o(1).$$

When truncation is mild, this is approximately $\log_2(4.133\,\sigma)$, whereas uniform support width is about $2C$. The statistical effective width can therefore fall from roughly $2C$ to 4.133σ. With weak concentration the norm-ball bound may be looser than the interval bound; entropy itself is still no larger than uniform support entropy.

For illustration take the earlier hard bound $|E| < 2^{20}$, so $C = 2^{20} - 1$, $W = 2^{21} - 1$, and $q = 2^{64}$. Uniform entropy is approximately 21 bits and its support-based one-sample choice uses $r = 23$. These are numerical finite-sum evaluations, not runs of a tensor solver. Here $r_V = \lceil \log_2(4A_V) \rceil$, so the false-candidate term is at most $R_n(b)2^{-n}$; true rejection is added separately. Without a certified parity-bias bound use $R_n(1) = n/2$. For a truncated Gaussian the appropriate bias is the finite-sum quantity $|\sum_{z=-C}^{C}(-1)^z e^{-\lambda z^2}|/Z_C(\lambda)$; the untruncated Gaussian approximation is not a substitute for this value.

C/σ_0	$H(E)$	Entropy saved	$r_{1.1v}$	r_{4v}
3	20.439	0.561 bits	23	24
6	19.462	1.538 bits	22	23
10	18.725	2.275 bits	21	22
20	17.725	3.275 bits	20	21

Reproduce the entries by setting $p_z = e^{-z^2/(2\sigma_0^2)}/\sum_j e^{-j^2/(2\sigma_0^2)}$, computing $v = \sum_z p_z z^2$ and $H = -\sum_z p_z \log_2 p_z$, then applying (12.8). At $C/\sigma_0 = 6$, the numerically optimized rate $I(1.1v)/\ln 2$ is about 0.00338, whereas $I(4v)/\ln 2$ is about 1.168. This shows why a smaller retained modulus and the same reliability exponent cannot be inferred from entropy savings alone. These floating-point illustrations are not interval-certified probability bounds; rigorous certification uses the finite-sum formulas with controlled numerical errors.

At $C/\sigma_0 = 3$ the conservative $4v$ norm test even uses one more bit than the support-only method. An attacker can keep the 23-bit support test. At stronger concentration the norm test can save bits, but its additional network constants still enter a crossover calculation. Entropy bits saved are not cryptographic security bits saved.

12.8 Approximate samplers, asymptotic families, and BKZ

An implementation may use a rounded, tabulated, or otherwise approximate Gaussian law. The norm proof requires its *actual* second moment and a valid tail bound, not exact Gaussian probabilities. Every independent bounded law satisfies the bounded-variable concentration estimate above. For a known approximate law, its actual moment-generating function can give a sharper rate. Closeness to a Gaussian must include an approximation metric and accuracy if Gaussian-specific numerical rates are imported. If per-coefficient total variation distance is ϵ, the joint laws of n independent coefficients have distance at most $n\epsilon$: couple each pair to agree with probability at least $1-\epsilon$ and union-bound disagreements. A fixed small ϵ does not justify negligible whole-vector error as n grows. Direct analysis of the implemented law avoids that issue.

For fixed C/σ and fixed positive reliability slack, replacing uniform noise by a concentrated law chiefly changes precision and constants. Since $\log \sigma = \log W + O(1)$ in the growing-width regime, it does not change the order of the subexponential criterion. With $V = (1+\eta)\sigma^2$, fixed $\eta > 0$, and adequate modulus, one may use $r = O(\log(\sigma+2))$. The norm-based bound is subexponential when

$$\log(\sigma+2) = o(\sqrt{\log n}),$$

provided true acceptance stays high along the family. For a growing ratio C/σ, the elementary bounded-support concentration exponent may deteriorate; a uniform Gaussian-like exponential-moment bound is then needed, rather than merely a variance. A large support populated by rare tails need not dictate the retained width for binary-secret recovery. Noise residues may wrap without losing the binary secret's lift.

The BKZ comparison must change too. At fixed weights the expected error contribution to the primal target norm is $\ell w_e^2 v$, which decreases as the truncated Gaussian concentrates. This benefits the conventional short-target strategy as well. Its samples still have original modulus q, not the retained D used by tensor search. Reoptimize that baseline with the new error law; improving our statistical threshold does not prove greater relative advantage over BKZ. For fixed q and fixed nonzero variance, the energy hypotheses in Proposition 6.1 continue to hold for typical inputs, so its explicit model still has linear block sizes. Its limitations and the solver cutoff remain unchanged.

There is a separation: unchanged support tests give unchanged guarantees; distribution-aware tests exploit lower entropy and energy; their encoding retains the asymptotic size bound; and concrete advantage requires both new constants and a new original-modulus baseline calculation.

12.9 Residue support can matter independently of Gaussian shape

If every error coefficient has a known parity, reduction modulo two removes that noise after subtracting the known parity vector. When the public multiplier is a unit modulo two, inversion recovers a binary secret immediately. Errors uniform in $\{-1,+1\}$ illustrate this elementary issue. Uniform intervals and ordinary wide discrete Gaussians usually occupy both parities and lack this deterministic leak. Calling a law Gaussian-like does not replace analysis of its residue support, independence, and concentration.

13 Conclusions and a checkable interpretation

The initial power-of-two Ring-LWE case illustrates the conditional method through a short dependency chain:

1. The power-of-two negacyclic ring is a finite chain ring.

2. Every nonzero binary difference generates an ideal containing $2R_D$.
3. Random multiplication is uniform on that ideal, giving the stated collision probability for interval errors.
4. A triadic, single-split transform gives an exact, uniquely extended, $O(nr)$-edge multiplication verifier.
5. The assumed tensor solver counts and searches that verifier within $2^{O(mnr/\sqrt{\log(mnr)})}$ time and polynomial space.

For $q \geq 4W$, one ring sample and $r = \lceil \log_2 W \rceil + 2$ suffice for failure at most 2^{-n}. The derived time is subexponential in n when $\log W = o(\sqrt{\log n})$. With larger widths or many needed samples, the relevant quantity is mr rather than r alone.

Uniform interval errors admit the exact first moment of Theorem 3.6. For one sample its leading bound is $(2W/D)^n R_n(\epsilon_W)$ with $R_n \to 1$; the general multi-sample formula retains $A_j(\epsilon_W)^m$, so its parity correction cannot be removed by the same simplification. The compact binary encoding of Section 4.6 needs no generic bipartite subdivision and gives proved smaller size certificates with their interface and parameter restrictions explicit.

For Gaussian-like noise, support-only guarantees remain available. Distribution-aware tests exploit lower entropy and energy. The pointwise acceptance-set bound is $(n/2)2^n|\mathcal{T}|/D^n$; Theorem 12.2 replaces $n/2$ by $R_n(b)$ after averaging independent noise with parity bias bounded by b. For fixed $b < 1$ this is $1 + o(1)$, not necessarily at most one at finite dimension. The sufficient condition $r > H(E \bmod D) + 1$ with strict slack matches the entropy converse asymptotically. The norm verifier is $O(nr)$, but its additional edges and both failure terms must be counted. Fixed-ratio Gaussian concentration chiefly changes constants, and lower variance also benefits the original-modulus conventional baseline. No concrete crossover follows merely from entropy bits or smaller certificates.

The conclusion for the requested generalizations is therefore not a list of unconditional impossibility statements:

Change	What the proof establishes or still needs
Other moduli	An adequate two-part works immediately. Small odd factors, or suitable products of them, also admit positive results. Large growing factors leave a word-width overhead.
Other rings	Monic polynomial reduction has a compact verifier. Recovery requires an image-size argument; irreducibility modulo two is a positive case, while a cyclic example defeats the original containment lemma.
Other secrets	Bounded alphabets work directly. Arbitrary secrets can be eliminated in favor of bounded errors with a unit base and extra samples.
Dense LWE	The public map lacks the proved convolution encoding. The direct verifier is too large; no general subexponential LWE theorem follows.
General reduction	Projected SVP blocks do not retain the required ring and coefficient promises. A generic BKZ speedup needs a new oracle theorem.

For the original structured family, the correct overall attack estimate is the minimum of the direct conditional method after reduction to D and all applicable conventional attacks on the original modulus q. Discarding high bits may make the conventional problem harder; comparing only against that reduced problem could overstate the improvement. BKZ's internal sieving exponent does not change merely because this alternative exists. Proposition 6.1 proves that the specified primal GSA test predicts asymptotically full

block sizes for fixed q and positive secret and error energy, even after optimizing its diagonal weights. Our subexponential bound eventually beats that modeled exponential sieve cost. This is an implication of explicit heuristics, not a lower bound for actual BKZ, and the fixed-modulus embedding has structural limitations at unbounded dimension. A genuinely subexponential conventional baseline would require a separate comparison; none is exhibited here for the intended fixed family. The solver's cutoff and unknown constants preclude a concrete security revision.

The audit trail below lists the exact invariants and source snapshots needed for an independent mathematical or machine-assisted review. Numerical toy checks are included as verification aids, not as substitutes for the proofs.

A Verification roadmap and reproducible checks

A.1 Interfaces and invariants to audit

An independent checker can audit the manuscript in the following order. No task in the list requires trusting a numerical lattice estimator.

1. Verify that all coordinates use the power basis and that n is an even power of two. Distinguish the ciphertext modulus q, reduced modulus D, and tensor-counting precision 2^P; they serve different roles.

2. In Lemma 2.1, check the characteristic-two factorization, the unit-lifting geometric series, the relation $2 = u(X+1)^n$, and the ideal-chain proof. The latter is needed, not merely locality of the ring.

3. In Lemma 3.2, check the denominator $W(D/2)$ and the numerators $N_0^2 + N_1^2$ and $2N_0N_1$. The averaging uses independent uniform error coordinates. Use the support-only lemma instead if the error distribution is not uniform.

4. Check that independence is used between different samples, while different candidate secrets need only a union bound. No independence between candidate explanations is assumed.

5. Verify the principal-root identity in the auxiliary ring, the invertibility of the odd transform length, and both no-wraparound bounds. An element of exact order L alone would not be enough over a ring with zero divisors.

6. Verify the two-stage indices in (4.3)–(4.4). Only one intermediate array is present in each transform. Replacing this by a recursive FFT circuit changes the wire count.

7. Count $2Jr$ Boolean wires per auxiliary word and check the terminal fanin of at most four scalar coefficients. There is no hidden copying to all frequencies outside the transform predicates.

8. Check canonical coefficient representations and the forward determination of all arrays. Count secrets, not unconstrained arithmetic witnesses. At modulus 2^r every r-bit scalar representation is canonical.

9. Check bipartiteness and apply subdivision only when needed. Apply the degree-cap size parameter before calling the source theorem. Exact count precision must exceed the number of potentially free Boolean edges.

10. Compare the direct attack to an explicit BKZ schedule, not to a standalone sieve exponent with dimension n substituted for β. Finally check the source algorithm's starting range separately from the asymptotic exponent.

11. For Gaussian-like noise, verify the exact binary valuation counts in Lemma 3.5, rather than charging the worst image size to every candidate. Keep the true-rejection term and false-candidate term separate in (12.9). Check the actual implemented noise law before importing a Gaussian Chernoff exponent.

A.2 Reference pseudocode for the algebraic verifier

The following is reference pseudocode for the conservative witness arrays. The compact binary construction eliminates interfaces as described in Section 4.6; it uses the same transform identities. It is *not* an implementation of the assumed contraction algorithm. Each line becomes one layer or a collection of local predicates as proved in Section 4.

1. $J \leftarrow$ smallest power of three with $J^2 \geq n$; $B \leftarrow \lceil n/J \rceil$; $L \leftarrow$ smallest power of three at least $2B - 1$.
2. $A \leftarrow (\mathbb{Z}/2^r\mathbb{Z})[Y]/(Y^{2J} + Y^J + 1)$; $\omega \leftarrow Y^{3J/L}$.
3. Pack a, u into B polynomials of degree below J, using zero padding up to L words of A.
4. Compute $\widehat{a}$ publicly. For the witness, compute the two forward layers for u using (4.3) and (4.4).
5. $z_t \leftarrow \widehat{a}_t \widehat{u}_t$ in A for every t.
6. Apply the same two layers with ω^{-1} and multiply the second layer by L^{-1} modulo 2^r.
7. Interpret the result as the unique degree-below-$2J$ polynomials C_t, and recombine $\sum_t X^{tJ} C_t(X)$.
8. Fold degree $i + n$ into degree i with a minus sign, check the public affine relation, and check the output error interval.

To test the ring arithmetic directly, multiply ordinary polynomials by schoolbook convolution and reduce every scalar coefficient modulo 2^r. For auxiliary multiplication, reduce powers Y^d with $d \geq 2J$ by $Y^d = -Y^{d-J} - Y^{d-2J}$, processing degrees from high to low. This gives a simple exact reference independent of the Fourier factorization.

A.3 A small explicit chain-ring example

For $n = 2$, $X^2 = -1$, and $\delta = 1 + X$, multiplication gives

$$(a_0 + a_1 X)(1 + X) = (a_0 - a_1) + (a_0 + a_1)X.$$

Its image consists of the pairs whose coefficients have equal parity. It has $D^2/2$ elements and contains every even pair. Thus multiplication is not generally a permutation, but the containment used in our proof holds. With an interval of width three, N_0, N_1 are one and two in some order. Conditional on an even output difference, the scalar acceptance probability averaged over the true error is $10/(3D)$; for an odd difference it is $8/(3D)$. The larger equals α_3/D, matching Lemma 3.2.

A.4 Checks performed during preparation

Exact arithmetic tests checked the auxiliary multiplication and the actual two-stage transform, not only a black-box FFT. They included 512 exhaustive small multiplication cases, 28 additional cases up to $n = 512$, direct-DFT comparisons, inversion, every nonzero-frequency principal-root cancellation in the tested configurations, and the displayed wire and degree bounds. Separate rational-arithmetic tests checked the containment $HT \subseteq \delta T$

through inverses with odd coefficient denominators in 300 small examples. Small exhaustive-secret searches for binary and ternary inputs were consistent with the stated ambiguity bounds. Small CRT submodule box counts were checked as well.

The Gaussian-noise refinement was additionally checked in 516 exact binary-difference cases over $n = 2, 4$ and moduli $2, 4, 8$ where applicable. The tests enumerated multiplication images and compared their cardinalities with $D^n/2^j$, and checked the exact count 2^{n-j-1} of each valuation class for every secret in those cases. The entropy table in Section 12 uses its displayed finite-sum formulas.

The exact first moments were also checked by full enumeration at $n = 2, 4$, including odd and even interval widths. The enumerator recurrence was compared with direct valuation sums in exact rational arithmetic. Compact parameter choices were checked against schoolbook multiplication with J no longer restricted to a power of three. The compact recipe was checked in 232 edge/degree cases over powers 2^3 through 2^{60}, including the boundary of $r \le n^{3/2}$; 16 complete compact-parameter multiplications and 16 signed-forward checks used dimensions 8 through 1024. Analytic constants in Section 4.6 are proved there, rather than inferred from those finite tests.

These finite checks are useful for finding index, parity, and sign errors. They do not verify the assumed ETH theorems, establish asymptotic probabilities by experiment, or replace any proof above.

A.5 A useful direct counterexample to an invalid shortcut

If the reduced modulus is a divisor of an even interval width W and $D \mid W$, uniform interval noise becomes exactly uniform modulo D. Every reduced label is then uniform independently of the secret. Therefore one cannot keep only a constant number of low bits for a growing-width uniform-noise family without checking the information loss. The choice $D > W$ in the main algorithm is substantive, not cosmetic. Typical-set methods can exploit nonuniform residual laws, but cannot extract information from an exactly uniform one.

B Source assumptions, versions, and notation map

The three requested source PDFs were downloaded from the supplied GitHub repositories for this manuscript. The live `main` URLs can change. The following SHA-256 hashes identify the exact PDF bytes consulted; the two lines of each hash are concatenated.

Source	Version and SHA-256

OpenAI ETH	Revised 25 October 2026; Appendix D.1 is the tensor interface.
	`5b1612dc2eebb2cad5b4cbe62d1fed2982`
	`f01ab1908c26f70010a651487e8123`

Anthropic ETH	Dated 26 October 2026; main theorem is the SAT result.
	`7af518b09342c967ea371c74b8989806`
	`684c5d02aabe5763ca08554aff5025a0`

ECDLP	Version 0.2, 29 October 2026.
	`82e273629b437660607117d5e99f4001`
	`41ed7aa78b25c6749b03651f42deaa11`

The earlier conversation attachment was an earlier ECDLP snapshot. We consulted the requested repository's version 0.2 for the paper and verified the same relevant tensor interface. Our triadic polynomial-ring encoding is an extension proved in Section 4, not a theorem attributed to the ECDLP authors.

The original OpenAI tensor theorem uses a size parameter rather than necessarily the literal edge count, allows succinct tensors over a truncated polynomial ring modulo a power of two, and permits exact counting by increasing the precision. Our specialization uses Boolean predicates, no truncation variable, polynomial-time entries, and the explicit adjusted size parameter in Section 1. The Anthropic main theorem supplies an independent

SAT refutation; we do not assume it automatically contracts our more general succinct network. Neither source theorem is independently verified here, in accordance with the requested assumptions.

Symbol	Meaning in this manuscript
n	Ring degree and number of secret coefficients.
$q = 2^t$	Original ciphertext modulus and its bit exponent.
$D = 2^r$	Exact divisor used by the search; r is not a security level.
W	Number of integers in the known noise interval.
m	Number of ring samples used in the binary attack.
E, Δ	Actual Boolean edge count and maximum port degree.
$s_\star$	Adjusted size parameter passed to the tensor theorem.
d	Dimension of a primal lattice embedding, not generally n.
β	BKZ block size, determined by an attack model and parameters.
N	Number of SVP calls in a specified BKZ schedule.

When comparing formulas, the ECDLP manuscript's b is the bit size of its curve field; it is unrelated to our public label b or noise width. Similarly, the ETH papers' Boolean-variable counts are not ring degrees until an encoding-size theorem explicitly relates them.

References

[1] OpenAI. *The Exponential Time Hypothesis is False*. Preprint, 17 October 2026, revised 25 October 2026. https://github.com/openai/math/blob/main/preprints/The-Exponential-Time-Hypothesis-is-False-October-25-2026/paper.pdf.

[2] Claude, Anthropic. *A $2^{o(n)}$ Time Algorithm for 3-SAT*. Preprint, 26 October 2026. https://github.com/anthropics/formal-math/blob/main/eth/paper.pdf.

[3] Akshima, R. Bacho, Y.-H. Huang, E. Kiltz, A. Nishaburi, S. Nooripoor, N. Tittelbach, and E. Wiedijk. *A Subexponential Time Algorithm for Elliptic Curve Discrete Logarithm*. Version 0.2, 29 October 2026. https://github.com/bochum-crypto/subexponential-ecdl/blob/main/subexponential-ecdl.pdf.

[4] C.-P. Schnorr. A hierarchy of polynomial time lattice basis reduction algorithms. *Theoretical Computer Science* 53(2–3), 201–224, 1987. https://doi.org/10.1016/0304-3975(87)90064-8.

[5] Y. Chen and P. Q. Nguyen. BKZ 2.0: Better lattice security estimates. In *ASIACRYPT 2011*, LNCS 7073, 1–20. Springer, 2011. https://doi.org/10.1007/978-3-642-25385-0_1.

[6] A. Becker, L. Ducas, N. Gama, and T. Laarhoven. New directions in nearest neighbor searching with applications to lattice sieving. In *SODA 2016*, 10–24. SIAM, 2016. https://doi.org/10.1137/1.9781611974331.ch2. Author manuscript: https://ir.cwi.nl/pub/23692/23692.pdf.

[7] D. G. Cantor and E. Kaltofen. On fast multiplication of polynomials over arbitrary algebras. *Acta Informatica* 28, 693–701, 1991. https://doi.org/10.1007/BF01178683.

[8] A. Pospelov. Faster polynomial multiplication via discrete Fourier transforms. *arXiv:1010.1101*. https://arxiv.org/abs/1010.1101.

[9] H. Q. Dinh. Negacyclic codes of length 2^s over Galois rings. *IEEE Transactions on Information Theory* 51(12), 4252–4262, 2005. https://doi.org/10.1109/TIT.2005.859284. The ring structure used here is independently proved in Lemma 2.1.

[10] S. Arora and R. Ge. New algorithms for learning in presence of errors. In *ICALP 2011*, LNCS 6755, 403–415. Springer, 2011. https://doi.org/10.1007/978-3-642-22006-7_34.

[11] A. Jain, H. Lin, and S. Saha. A systematic study of sparse LWE. IACR Cryptology ePrint Archive, Report 2024/1589, 2024. https://eprint.iacr.org/2024/1589.

[12] Lattice Estimator contributors. *LWE Primal Attacks*. Official documentation, accessed 30 October 2026. https://lattice-estimator.readthedocs.io/en/latest/algorithms/lwe-primal.html.

[13] Lattice Estimator contributors. *estimator.simulator.ZGSA*. Official documentation, accessed 30 October 2026. https://lattice-estimator.readthedocs.io/en/latest/_modules/estimator/simulator.html#ZGSA.

[14] OpenAI. *The Exponential Time Hypothesis is False: scope of the Lean formalization*. Official repository documentation, accessed 21 November 2026. https://github.com/openai/math/blob/main/lean/docs/378.md.