ECC RESEARCH PROGRAM · P3.2

No Single Constant: Parameterizing the Quantum Cost of the CSIDH Class-Group Action

A verified toy isogeny action, a finite-range sieve fit, a fully named physical-cost calculator, and a conditional query lower bound

math.st [1] **@math__street** [1]

with GPT 5.6 Sol · Claude Fable 5

[1]Independent research collective, math.st · July 2026

ABSTRACT

The best quantum attacks on CSIDH-style commutative group actions run in time $\exp(c\sqrt{\log N})$ in the class-group order N, and the problem posed as P3.2 asks for two things: the leading constant and lower-order terms of that cost under an explicit physical error-correction model, and a quantum query lower bound for group-action inversion. We report the honest outcome of one focused research effort: **neither formal task was completed**, and we document precisely why, together with the validated partial artifacts that survive. We verify a toy CSIDH action exhaustively — at $p = 59$ and $p = 419$ the explicit rational Vélu orbits match the reduced-form class numbers 9 and 27 and the generated permutation actions are abelian and regular. We simulate a **simplified** fixed-batch low-bit-collision sieve over $N = 2^n$, $24 \le n \le 96$ (1,000 seeded trials), and fit $\ln Q = c\sqrt{\ln N} + d \ln\ln N + k$, obtaining $c = 2.68677$ with trial-bootstrap 95% CI $[2.65454, 2.71923]$ — a finite-range constant for that named schedule only, **not** a new constant for Kuperberg's collimation sieve. We build a cost calculator with zero hidden numeric defaults that reproduces two published logical endpoints exactly ($2^{71.6}$ and 2^{56} T gates) while keeping every surface-code output explicitly illustrative, and we rank physical-cost sensitivity over named parameter endpoints. On the lower-bound side we prove, by a dimension argument, that m independent standard phase-state queries recover a hidden shift with probability at most $2^m/N$, hence $\Omega(\log N)$ queries are necessary — conditionally on that query interface, which provably does not cover an arbitrary coherent call to a structured action circuit. The negative conclusion is itself the finding: a "physically meaningful universal constant" is not defined until the sieve schedule, oracle circuit, memory technology, and error-correction layer are all fixed by named assumptions, and we exhibit exactly where each published estimate makes a different choice.

Keywords. CSIDH · class-group action · hidden shift · Kuperberg sieve · quantum resource estimation · surface code · query lower bounds

1 Introduction

CSIDH [1] instantiates a commutative group action: the ideal class group $\mathrm{Cl}(\mathcal{O})$ of an imaginary-quadratic order acts freely and transitively on a set of supersingular elliptic curves over a

prime field. Its conjectured quantum security rests on the abelian hidden-shift problem, for which the Kuperberg family of sieves [2], [3], [4] gives subexponential algorithms of shape $\exp(c\sqrt{\log N})$ in the group order N, applicable to isogeny computation through the reduction pipeline of Childs, Jao, and Soukharev [5]. Unlike the situation for RSA or ECC, where concrete quantum resource estimates broadly agree, published CSIDH-512 attack costs disagree by orders of magnitude: Bonnetain and Schrottenloher report $2^{71.6}$ logical T gates [6], Peikert reports an optimistic-oracle range near 2^{56} to 2^{60} [7], and Bernstein, Lange, Martindale, and Panny count $\approx 2^{40}$ nonlinear bit operations for a **single** oracle call at failure probability below 2^{-32} [8].

The problem statement for P3.2 therefore asks two formal questions. Let N denote the class-group order. **Task 1:** under an explicit physical error-correction model, determine the leading constant and lower-order terms in a cost of the form $\exp(c\sqrt{\log N})$ for Kuperberg-style abelian hidden-shift algorithms. **Task 2:** prove a quantum query lower bound for group-action inversion.

Main finding (honest outcome). PROVED **Neither formal task was completed; the research outcome is recorded as FAILED.** No physically calibrated universal constant and no unrestricted structured-action query lower bound was obtained. What survives is a set of validated partial artifacts: (i) an exhaustively verified toy class-group action whose explicit Vélu orbits match reduced-form class numbers 9 and 27; (ii) a finite-range fitted constant $c = 2.68677$, 95% CI $[2.65454, 2.71923]$, valid **only** for one named simplified sieve schedule over $24 \le \log_2 N \le 96$; (iii) a physical-cost calculator with no hidden numeric defaults that reproduces the published logical endpoints $2^{71.6}$ and 2^{56} exactly; and (iv) a conditional $\Omega(\log N)$ query lower bound whose interface provably does not extend to arbitrary coherent structured-action queries without a new reduction. The analysis isolates **why** no single constant exists in the current literature: every published number fixes a different sieve schedule, oracle model, memory technology, and error-correction layer.

1.1 Contributions and honest scope

We contribute (i) an exhaustive verification methodology and results for toy CSIDH actions at $p = 59$ and $p = 419$ (§3, Figure 1); (ii) a transparent simplified sieve simulator with a three-term fit and trial-bootstrap uncertainty (§4, Figure 2); (iii) a fully parameterized logical-to-physical cost calculator, two exact published-endpoint reproductions, and a sensitivity ranking (§5–6, Figure 3, Figure 4); (iv) a conditional phase-state query lower bound with a precise statement of its boundary (§7); and (v) an assumption-by-assumption account of why published CSIDH attack costs disagree (§8).

The scope rules of the underlying research repository are strict, and we keep them. PROVED Classical simulation can measure the implemented sieve's combinatorial counters, but it cannot measure quantum wall-clock cost; this is a definitional distinction between the simulator and a quantum execution. PROVED The simplified low-bit-pair simulator is **not** Kuperberg's or Peikert's full phase-vector collimation sieve, and the surface-code outputs below are illustrative, not published CSIDH physical estimates. Every claim in this paper carries the epistemic tag under which it was recorded in the research log; untagged sentences are exposition.

2 Setting and notation

Let $\mathcal{O}$ be an imaginary-quadratic order of discriminant $\Delta < 0$ and $\mathrm{Cl}(\mathcal{O})$ its ideal class group, of order $N = h(\Delta)$. CITED In CSIDH, $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$ acts freely and transitively on the set of supersingular curves over $\mathbb{F}_p$ with $\mathbb{F}_p$-rational endomorphism ring $\mathbb{Z}[\sqrt{-p}]$, and small split primes ℓ act through horizontal ℓ-isogenies computable by Vélu's formulas [1].

Definition 1 (hidden shift and group-action inversion). For a cyclic model group $\mathbb{Z}_N$, the **hidden-shift problem** is: given oracle access to injective $f_0, f_1 : \mathbb{Z}_N \to X$ with $f_1(g) = f_0(g + s)$, recover s. **Group-action inversion** is: given x and $y = s \cdot x$ for a free transitive abelian action, recover s. Kuperberg-style algorithms solve the first problem with $2^{O(\sqrt{\log N})}$ resources [2], [4]; the second reduces to the first (§7, Proposition 9).

Throughout, Q denotes an abstract count of oracle queries (phase states consumed), $n = \log_2 N$, and ln is the natural logarithm. "The calculator" means the JSON-driven cost model of §5, whose every numeric input is a named serialized assumption.

CITED The asymptotic statements of [2], [3], [4] do not select a physical gate set, error-correcting code, QRACM implementation, or finite-size leading constant; these are simply not variables in the theorems. Task 1 is precisely the demand that they become variables, and §8 shows the published literature answers it in mutually incompatible ways.

3 An exhaustively verified toy class-group action

Before measuring any sieve, the research validated the object being attacked: an explicit, fully enumerable class-group action built from the same ingredients as CSIDH.

3.1 Construction

For $p \equiv 3 \pmod 4$, the curve $E_0 : y^2 = x^3 + x$ over $\mathbb{F}_p$ has trace zero. PROVED $\#E_0(\mathbb{F}_p) = p + 1$ for $p > 3$: pairing each x with the quadratic-character sum forces trace zero, and the implementation additionally checks the count exhaustively. The toy instance takes discriminant $\Delta = -4p$, enumerates all primitive reduced positive-definite binary quadratic forms of that discriminant to obtain $h(-4p)$ independently of any isogeny computation, and then walks the isogeny graph: for each odd split prime $\ell \in \{3, 5, 7\}$ with $\ell \mid p + 1$, the rational ℓ-isogeny steps are computed by explicit Vélu quotients, and curves are canonicalized up to $\mathbb{F}_p$-isomorphism **retaining twist information** rather than identifying by j-invariant alone. HEURISTIC The working hypothesis of attempt A001 was that small split-prime steps generate the full class-group orbit in a suitably chosen instance; the refuting test was an orbit smaller than the reduced-form class number or a generator step that fails to be a permutation.

Lemma 2 (regularity criterion). PROVED A finite group action is free and transitive if one orbit contains every point and every point stabilizer is trivial. Moreover, for a finite group action, an orbit of size equal to the acting group's order has trivial stabilizer.

Proof. The first sentence is the definition of free and transitive actions applied pointwise. For the second, the orbit-stabilizer theorem gives $|G| = |\text{orbit}| \cdot |\text{stab}|$; if $|\text{orbit}| = |G|$ then $|\text{stab}| = 1$. The implementation instantiates this with the computed permutation group: `regular = true` is set exactly when the generated group is transitive and the base-state stabilizer contains only the identity. □

3.2 Results

EMPIRICAL: p = 59, degrees 3, 5 Reduced-form enumeration gives $h(-236) = 9$; the explicit Vélu orbit has 9 prime-field isomorphism classes, and the generated permutation group is abelian and regular. EMPIRICAL: p = 419, degrees 3, 5, 7 Reduced-form enumeration gives $h(-1676) = 27$; the explicit Vélu orbit has 27 classes, the three generators commute, the action is transitive, and the base-state stabilizer is trivial — orbit size, class number, and action-group order all equal 27. Table 1 and Figure 1 summarize both instances.

Table 1: EMPIRICAL: exhaustive enumeration, 2026-06-30 Toy CSIDH actions verified end-to-end. Class numbers come from independent reduced-form enumeration; orbits from explicit rational Vélu quotients. Data: `verify_toy_action_p59_20260630.json`, `verify_toy_action_p419_20260630.json`.

Instance	Δ	degrees ℓ	$h(\Delta)$	orbit	group order	regular
$p = 59, \#E_0(\mathbb{F}_p) = 60$	-236	3, 5	9	9	9	yes
$p = 419, \#E_0(\mathbb{F}_p) = 420$	-1676	3, 5, 7	27	27	27	yes

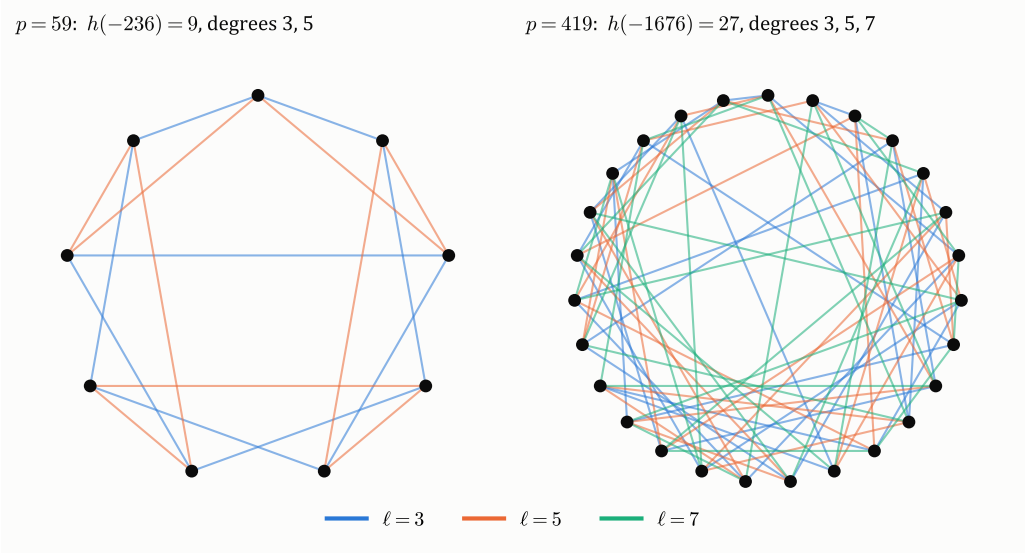

Figure 1: EMPIRICAL: p = 59, 419 The two verified orbits as transition graphs: each node is an $\mathbb{F}_p$-isomorphism class of curves, each chord an explicit rational ℓ-isogeny step. Both graphs are single orbits of commuting permutations with trivial base stabilizer — a regular abelian action, as the CSIDH theorem [1] predicts. Drawn from the serialized transition tables in the `verify_toy_action_*.json` artifacts.

Two honesty notes. EMPIRICAL: A001 prediction test at p = 419 The pre-registered prediction was that degrees $\{3, 5, 7\}$ might generate only a proper subgroup; it was **refuted** — the generated action has the full 27 elements. EMPIRICAL: first p = 419 twist fixture An initial twist-separation test built on the $j = 1728$ starting curve was invalid: at that exceptional automorphism class the nominal quadratic twist is already prime-field isomorphic, so the fixture could not separate twists. The corrected test uses a generic Vélu quotient. We record the failed fixture because silent fixture repair is exactly how toy validations become overclaims.

PROVED The toy experiment retains the prime-field trace-zero action structure of CSIDH but uses short-Weierstrass representatives and tiny discriminants; it is not a parameter-level reproduction of CSIDH-512.

4 A simplified sieve and its finite-range constant

Task 1 asks for the constant of a Kuperberg-style algorithm. The research implemented and measured a deliberately simplified schedule whose combinatorics are fully transparent, as a calibration target for later, more faithful simulators — and the fitted constant below belongs to **this schedule only**.

Definition 3 (fixed-batch low-bit-collision schedule). Fix $N = 2^n$, a list multiplier $\mu = 8$, a block-width scale $\sigma = \sqrt{2}$, combination success probability $q = 1/2$, and target probability $t = 1/2$. Set the block width $b = \max\left(1, \operatorname{round}\left(\sigma\sqrt{n-1}\right)\right)$ and partition the lowest $n-1$ label bits into S stages of width at most b. The list size is $L = \mu \cdot 2^b$. Each stage throws L labels uniformly into 2^w buckets (multinomial occupancy), forms the exact number of disjoint same-bucket pairs $\sum_j \lfloor c_j/2 \rfloor$, and thins them binomially with success probability q; the final batch retains useful target labels binomially with probability t. Writing $\hat{r}_j$ and $\hat{s}_j$ for the observed pair and success rates at stage j, the abstract query count is back-propagated as

$$Q = \frac{L}{\hat{t}} \prod_{j=1}^{S} \frac{1}{\hat{s}_j}, \tag{1}$$

and combination attempts accumulate as $C = \sum_j \hat{r}_j \cdot (\text{required input at stage } j)$.

Remark 4 (what the counters are). PROVED The simulator's counters are classical data about an abstract sieve schedule: `query_count` is a back-propagated count of input phase states, not elapsed time, not a gate count, and not a physical resource estimate. The simulator contains no quantum state and no error-correction model. PROVED No simulated wall-clock timing is interpreted as quantum runtime anywhere in the result files.

4.1 The fit

EMPIRICAL: N = 2^n, 24 <= n <= 96, 10 sizes, 100 trials/size, seed 20260722 The simulator produced 1,000 query-count and combination-count observations. Fitting the per-size means of $\ln Q$ to

$$\ln Q = c\sqrt{\ln N} + d \ln\ln N + k \tag{2}$$

by least squares, with 2,000 within-size trial-bootstrap replicates (seed 20260723) for the confidence intervals, gives the parameters of Table 2 the fitted curve, all 1,000 trials, and the per-size residuals appear in Figure 2.

Table 2: EMPIRICAL: 24 <= n <= 96, 1,000 trials, 2,000 bootstrap replicates Three-term fit of the simplified schedule's query count. Data: `fit_sieve_n24-96_20260630.json`.

Parameter	Estimate	Bootstrap 95% CI
c (coefficient of $\sqrt{\ln N}$)	2.68677	$[2.65454, 2.71923]$
d (coefficient of $\ln\ln N$)	-0.93996	$[-1.04210, -0.83629]$
k (intercept)	4.85231	$[4.67444, 5.03096]$
log-cost residual RMSE	0.36454	—

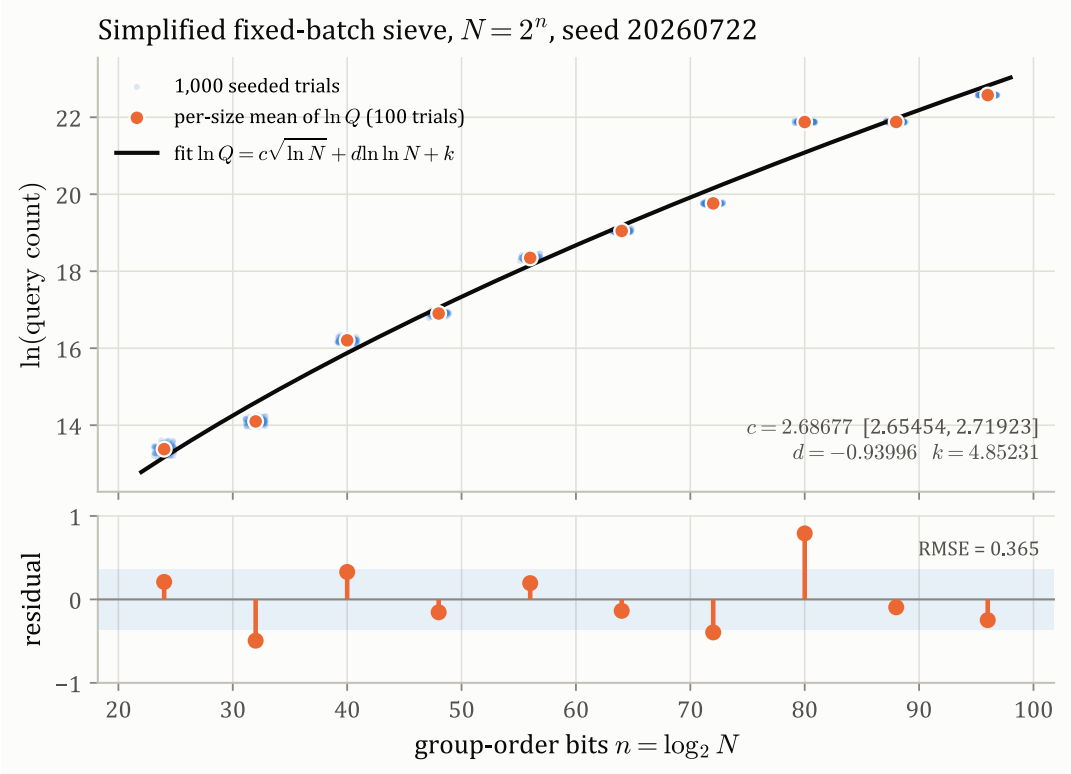

Figure 2: EMPIRICAL: seed 20260722 Simulated query counts for $N = 2^n$, $24 \le n \le 96$: 1,000 individual trials (light), per-size means (orange), and the three-term fit of Equation 2. The residual panel shows the per-size fit residuals against the ±RMSE band; the visible staircase structure reflects block-width rounding, which the bootstrap CI deliberately does **not** absorb. Data: `simulate_sieve_n24-96_seed20260722_20260630.csv`, `fit_sieve_n24-96_20260630.json`.

EMPIRICAL: same ten geometric means Model-shape comparison on the same data: the $\sqrt{\ln N} + \ln\ln N$ model attains in-sample RMSE 0.3645, against 0.3667 for $\sqrt{\ln N}$ alone, 0.4485 for a linear-log model, and 0.4988 for a log-log model. The square-root-log shape is preferred, but the margin over pure $\sqrt{\ln N}$ is small at these sizes.

4.2 What the fitted constant is, and is not

PROVED The reported CI quantifies seeded occupancy and measurement variation only: the bootstrap resamples trials within fixed sizes, so schedule discretization, block-width rounding, alternative sieve designs, and extrapolation error are all **excluded** from the interval by construction. PROVED No fit outside the sampled range $24 \le n \le 96$ is reported as empirical evidence.

CITED For calibration of expectations only: Kuperberg's collimation sieve has heuristic time $\tilde{O}\left(2^{\sqrt{2\log_2 N}}\right)$ for its optimized schedule [4], whose exponent in the natural-log units of Equation 2 has leading coefficient $\sqrt{2\ln 2} \approx 1.18$. PROVED The fitted $c = 2.68677$ is **not comparable** to that number and is not a new constant for Kuperberg's collimation sieve or for CSIDH as a whole: it belongs to a different, simplified schedule and counts a different resource (back-propagated phase states under fixed batching, not optimized quantum time). The gap between these two exponents is a measure of how much the constant in Task 1 depends on schedule fidelity — which is precisely why the task remained unsolved (§9).

5 A physical-cost calculator with no hidden defaults

The second ingredient Task 1 demands is a physical layer. The research position, enforced in code, is that every physical number is an **assumption to be named**, never a constant to be baked in.

Definition 5 (phenomenological cost map). Given sieve counters (Q, C, L) (queries, combination attempts, list size) and a configuration naming per-operation costs, architecture, and surface-code parameters, the calculator computes logical totals

$$T = Qt_{\text{orc}} + Ct_{\text{cmb}}, \quad D_{\text{ser}} = Q\delta_{\text{orc}} + C\delta_{\text{cmb}}, \quad D = \frac{D_{\text{ser}}}{Wu}, \tag{3}$$

with analogous Clifford and measurement counts, where W is the number of parallel workers and u their average utilization. Data logical qubits are $Q_{\log} = Q_{\text{base}} + L \cdot Q_{\text{phase}} + W \cdot Q_{\text{work}}$; the fault-location union bound is $F = (T + T_{\text{Cl}} + M) + Q_{\log} \cdot D$. The code distance is the least allowed odd d with

$$p_L(d) = A\left(\frac{p_{\text{phys}}}{p_{\text{th}}}\right)^{(d+1)/2} \leq \frac{B}{F}, \tag{4}$$

where A is the suppression prefactor, p_{th} the threshold, and B the failure budget. Physical qubits, runtime, and spacetime volume follow by direct substitution: data qubits $a_{\text{data}} Q_{\log} d^2$, factory qubits $n_{\text{fac}} a_{\text{fac}} d^2$, runtime $\max(D\gamma d, T/(n_{\text{fac}}\rho)) \cdot \tau_{\text{cycle}}$, and qubit-seconds as their product.

PROVED Every logical-operation, architecture, QRACM, and phenomenological surface-code number consumed by `calculate_cost` is serialized under `assumptions` in its output; the calculation function supplies no numeric defaults. PROVED QRACM bits are carried as a reported logical assumption and are never silently converted into surface-code data qubits — quantum memory, classical memory, and quantumly accessible classical memory remain separate metrics throughout, following the distinction the published estimates themselves insist on [4], [6], [7].

5.1 Exact reproduction of two published logical endpoints

A calculator that cannot reproduce published numbers under their own assumptions is not trustworthy under new ones. Two fixtures pin the logical layer to primary sources.

EMPIRICAL: published logical fixtures, 2026-06-30 The Bonnetain–Schrottenloher CSIDH-512 Section 3.3 row — 2^{19} queries at $2^{52.6}$ T gates per oracle call — is reproduced as $2^{19} \times 2^{52.6} = 2^{71.6}$ logical T gates, exact to ten decimal places in the base-two exponent [6]. The Peikert optimistic rounded endpoint — 2^{16} queries at 2^{40} T gates with 2^{40} QRACM bits — is reproduced as 2^{56} exactly [7]. PROVED The fixtures reproduce the cited multiplications, not the papers' full circuits or classical computations; and neither fixture thereby becomes a published **physical** estimate — each configuration labels its surface-code output illustrative.

Table 3: EMPIRICAL: calculator outputs, 2026-06-30 Three named configurations. The first two logical columns of the first two rows reproduce published endpoints exactly; **all** surface-code columns (distance, qubits, runtime) are illustrative outputs of the bundled phenomenological model and are not attributable to the cited papers. Data: `cost_model_n257_*_20260630.json`, `cost_model_n64_illustrative_surface_code_20260630.json`.

Configuration	queries (bits)	$\log_2$ T gates	QRACM bits	d	phys. qubits	runtime (s)
BS 2020 §3.3, $n = 257$ [6]	19.0	71.6	0	55	2.42×10^8	1.97×10^{17}
Peikert 2020 optim., $n = 257$ [7]	16.0	56.0	2^{40}	41	3.38×10^6	2.95×10^{12}
Illustrative sieve, $n = 64$	27.477	57.4	0	45	9.07×10^7	1.05×10^{13}

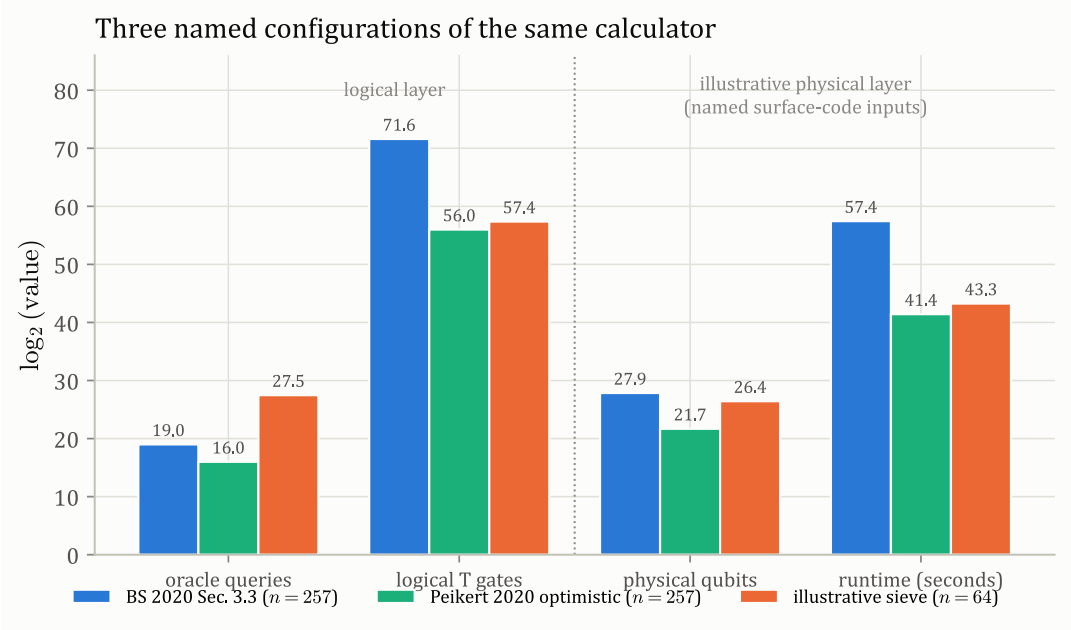

Figure 3: EMPIRICAL: calculator outputs, 2026-06-30 The three configurations of Table 3 in base-two logarithms. Left of the divider: logical quantities (the published endpoints live here). Right: physical quantities produced by the illustrative surface-code layer under its named assumptions. The same calculator, fed different named assumption sets, spans six orders of magnitude in runtime — the quantitative face of the "no single constant" finding.

EMPIRICAL: illustrative n = 64 configuration, 2026-06-30 For the simplified sieve at $n = 64$ under the illustrative configuration, the calculator reports 27.477 abstract query bits (geometric-mean $Q \approx 1.87 \times 10^8$ over 100 trials), code distance 45, 90, 679, 500 physical qubits, and runtime 1.05097×10^{13} seconds, with the logical-depth path (not magic-state production) as the bottleneck. None of these physical values is calibrated to a published CSIDH oracle; they are conditional outputs of the named JSON inputs by direct substitution.

6 Sensitivity: which assumption moves the answer most

If the physical answer is a function of named assumptions, the next honest question is which assumption dominates. The research ran a one-at-a-time sensitivity analysis over explicitly config-

ured endpoint ranges around the illustrative $n = 64$ base point (spacetime volume 9.53×10^{20} qubit-seconds).

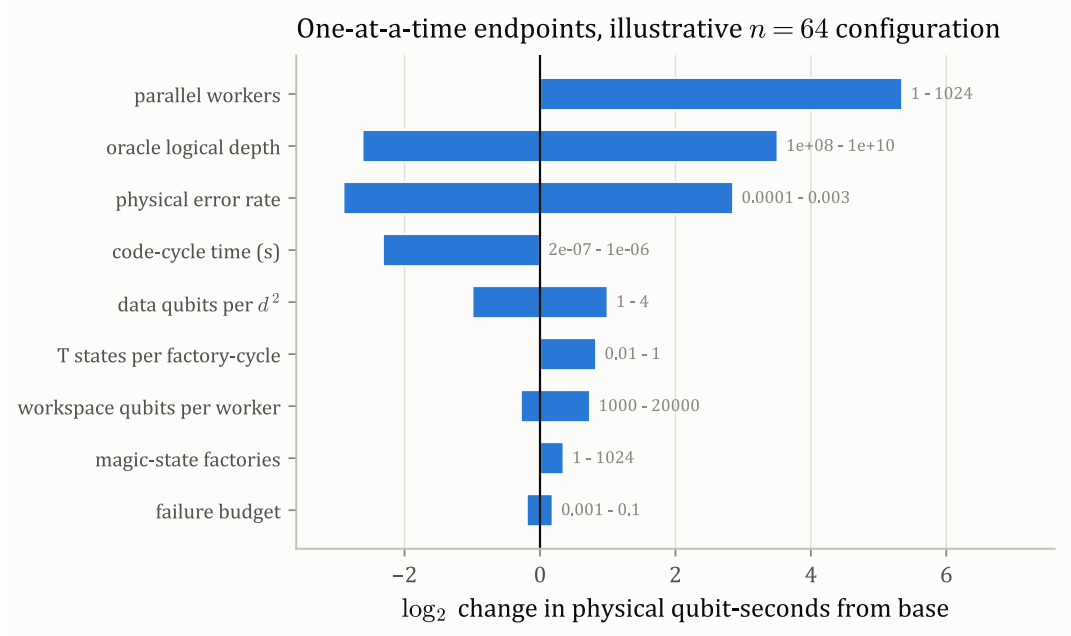

Figure 4: EMPIRICAL: configured one-at-a-time ranges, 2026-06-30 Tornado plot of $\log_2$ spacetime-volume change when each named parameter moves to its configured endpoints (annotated right), all others held at base. Parallelism and the oracle's logical depth dominate; the failure budget barely matters. Data: `sensitivity_n64_20260630.json`.

EMPIRICAL: configured one-at-a-time ranges, 2026-06-30 Parallel workers have the largest absolute effect on physical qubit-seconds over the deliberately broad supplied endpoints (1 to 1024), moving the metric by 5.350 bits; oracle logical depth ranks second at 3.510 bits, then physical error rate (2.905), code-cycle time (2.322), data qubits per d^2 (1.000), T-state production rate (0.830), per-worker workspace (0.740), factory count (0.350), and failure budget (0.197).

PROVED This ranking is conditional on the supplied endpoint ranges and is not an intrinsic global ordering of hardware importance: the method varies one named parameter at a time and cannot see interactions. Its value is diagnostic — it says where a future calibrated estimate must spend its precision (the oracle circuit and the parallelization model), and where precision is wasted (the failure budget).

7 A conditional query lower bound for group-action inversion

Task 2 asks for a quantum query lower bound. The research proved one — for a precisely delimited query interface — and proved that the delimitation is essential.

Definition 6 (independent standard phase-state queries). Fix $N \geq 2$ and let s be uniform in $\mathbb{Z}_N$. One query returns a classical label $a \in \mathbb{Z}_N$, distributed independently of s, together with the single phase qubit

$$|\psi_{a,s}\rangle = \frac{1}{\sqrt{2}}(|0\rangle + \exp(2\pi i a s/N)\ |1\rangle). \tag{5}$$

CITED This is the standard weak-Fourier-sampling state consumed by Kuperberg's cyclic hidden-shift algorithms ([4], Section 3).

Lemma 7 (dimension lemma). PROVED If a uniformly distributed label $s \in \{1, \ldots, N\}$ is encoded by density operators ρ_s supported on one common subspace of dimension at most d, then every POVM that guesses s has average success probability at most d/N.

Proof. Write Π for the projector onto the common support and $\{M_s\}$ for the POVM. Because each density operator satisfies $0 \preceq \rho_s \preceq \Pi$,

$$\frac{1}{N}\sum_s \operatorname{Tr}(M_s\rho_s) \le \frac{1}{N}\sum_s \operatorname{Tr}(M_s\Pi) = \operatorname{Tr}\frac{\Pi}{N} \le \frac{d}{N}. \tag{6}$$

□

Theorem 8 (phase-state query lower bound). CONDITIONAL: independent standard phase-state access Any quantum algorithm that recovers a uniform hidden shift $s \in \mathbb{Z}_N$ from m independent standard phase-state queries with success probability at least ε satisfies

$$m \ge \log_2 N + \log_2 \varepsilon, \tag{7}$$

and hence needs $\Omega(\log N)$ queries for constant success.

Proof. Under the stated condition, condition on all classical labels $a_1, \ldots, a_m$: these are independent of s, so afterward the shift is encoded entirely in the m returned qubits, whose joint state lives in a subspace of dimension at most 2^m. Lemma 7 bounds the average success of any measurement by $2^m/N$; rearranging $\varepsilon \le 2^m/N$ gives the bound. □

Proposition 9 (application to group-action inversion). PROVED For a free transitive cyclic action and endpoints $y = s \cdot x$, the maps $f_0(g) = g \cdot x$ and $f_1(g) = g \cdot y$ are injective and satisfy $f_1(g) = f_0(g + s)$; group-action inversion is therefore a hidden-shift instance. CONDITIONAL: a group-action query returns one standard phase state Under that interface convention, Theorem 8 applies to group-action inversion, and the bound is asymptotically tight at $\Theta(\log N)$ queries.

Proof. Injectivity is freeness; surjectivity onto the orbit is transitivity; and $f_1(g) = g \cdot (s \cdot x) = (g + s) \cdot x = f_0(g + s)$ by commutativity. Tightness follows from the cited upper bounds below. □

Remark 10 (tightness, and the query/time chasm). CITED Ettinger, Høyer, and Knill identify hidden subgroups of any finite group with polynomially many oracle calls in $\log N$ while allowing exponential total time [9], and later dihedral-coset work states an $O(\log N)$-query upper bound with exponential postprocessing [10]. CITED Bacon, Childs, and van Dam prove a sharper threshold for dihedral hidden-subgroup states: below one state per $\log_2 N$ bit the optimal identification probability is exponentially small; above it, constant [11]. The $\Omega(\log N)$ scale is therefore the **correct** answer for query complexity in this interface — and it is irrelevant to wall-clock security, since all known measurements at that query count take exponential time. Query lower bounds and time lower bounds are different tasks for this problem.

7.1 The boundary of the proof

Proposition 11 (why this does not resolve Task 2). PROVED The proof does not cover an arbitrary coherent query to a structured group-action circuit, because such a query need not be represented by one returned phase qubit. This is a mismatch between the proved interface and the stronger interface, not a missing algebraic step inside the dimension argument. PROVED Consequently the repository claims no superlogarithmic query lower bound; and no subexponential query lower bound can hold in any model that already admits the cited $O(\log N)$-query upper bound [9].

Extending Theorem 8 to the most general coherent structured-action oracle requires a precise oracle definition plus a reduction showing that arbitrary queries reveal no more shift information than the phase-state interface. That reduction is open (question Q012, §9), and it blocks the unrestricted wording of Task 2.

8 Why the literature does not define one constant

The failure of Task 1 is not an accident of effort: the published record itself does not determine a unique "physically meaningful" constant. Four independent modeling choices intervene between $\exp(c\sqrt{\log N})$ and a machine.

8.1 Sieve choice

CITED Kuperberg's first sieve, Regev's polynomial-space variant, Kuperberg's collimation sieve, and later subset-sum hybrids occupy **different** query/time/memory points rather than instantiating one universal constant [2], [3], [4], [10]. CITED Peikert reports CSIDH-512 examples near 2^{19} queries with 2^{32} QRACM bits and near 2^{16} queries with 2^{40} QRACM bits — selecting a memory point moves the query exponent by several bits within the **same** paper [7]. CITED Remaud, Schrottenloher, and Tillich list CSIDH-512 query exponents 19 for Regev's approach and 11 for one QRACM subset-sum tradeoff, each paired with much larger classical or quantum time exponents [10].

8.2 Oracle input distribution and failure budget

CITED Bernstein, Lange, Martindale, and Panny count $1,118,827,416,420 \approx 2^{40}$ nonlinear bit operations for one stated CSIDH-512 action algorithm at failure probability below 2^{-32}, and also report a $0.7 \cdot 2^{40}$ variant [8]. CITED Peikert explicitly treats the optimistic oracle cost as an assumption, notes that fewer oracle calls permit a larger per-call failure probability, and gives a logical T-gate range — 2^{56} to 2^{60} — rather than an error-corrected physical cost [7]. The oracle is a moving target: its cost depends on the exponent distribution it must serve and the failure it may tolerate, which in turn depend on the sieve consuming it.

8.3 Logical metric

CITED Bernstein et al. distinguish nonlinear bit operations, reversible Toffoli gates, T gates, and stored intermediate bits; their generic conversions multiply nonlinear operations by up to two for Toffoli gates and then by up to seven for T gates [8]. CITED Bonnetain and Schrottenloher's Section 3.3 row combines a 2^{19} query count with a $2^{52.6}$-T-gate oracle into $2^{71.6}$ logical T gates plus 2^{86} classical work and quantum memory below $2^{15.3}$ [6]. PROVED The Peikert fixture in §5 uses the rounded 2^{40} optimistic T-gate endpoint, not the Bernstein et al. nonlinear-operation integer,

because those metrics are not identical — conflating them silently changes the answer by small constant factors that matter at this precision.

8.4 Memory technology and error correction

CITED The published estimates separately use quantum memory, ordinary classical memory, and quantumly accessible classical memory; equating their bits or access costs changes the optimization problem itself [4], [6], [7]. PROVED At the physical layer, the bundled model exposes physical error rate, threshold, suppression prefactor, failure budget, code-cycle time, distance-dependent cycle cost, data and factory footprints, T-state production rate, power, parallel workers, and utilization as named JSON fields — twelve knobs, every one of which §6 shows can move the final answer, and none of which the asymptotic theorems constrain.

The conclusion we draw is structural, not rhetorical: **"the" constant in $\exp\left(c\sqrt{\log N}\right)$ is a function, not a number**, until a named point in this assumption space is fixed. The calculator is that function, made executable; what the research could not do — and what remains open — is to argue that any particular point deserves the name "physical."

9 Limitations and open problems

We list what is **not** established, faithfully to the research state.

- PROVED The simplified low-bit-pair simulator is not Kuperberg's or Peikert's full phase-vector collimation sieve; the state representation and recursion differ in the source code. The fitted $c = 2.68677$ is a finite-range constant of the simplified schedule only.
- PROVED The surface-code outputs are illustrative and are not published CSIDH physical estimates; no physical value in this paper is calibrated to a published CSIDH oracle circuit.
- PROVED No lower bound is proved for an arbitrary coherent call to a structured CSIDH action circuit; the conditional Theorem 8 covers the independent phase-state interface only.
- **Open question Q012.** Does the logarithmic phase-state lower bound extend to a structured action oracle? Resolving this requires fixing that oracle's input/output registers and either reducing every query to the phase-state interface or constructing a counter-algorithm that exploits the extra structure. Blocking for the unrestricted Task 2.
- **Open question Q013.** What finite-size constant does the **full** collimation sieve have under one physical model? The bootstrap interval of Table 2 omits block-rounding and schedule-model error, and the repository has not implemented Peikert's phase-vector collimation recursion or calibrated a fault-tolerant oracle circuit. Blocking for a universal or physically calibrated c.

The recorded next action is specific: reimplement the binary phase-vector collimation recursion from [7], validate it against one published simulation row from that paper, and only then compare its finite-size constant with the simplified schedule's — because the dominant scientific gap is algorithm fidelity, not another regression on the same finite-range staircase.

10 Conclusion

P3.2 asked for the physically calibrated constant of Kuperberg-style attacks on class-group actions, and for a query lower bound for group-action inversion. PROVED Neither was achieved, and the honest verdict recorded in the research ledger is FAILED. The salvage is nontrivial and, we believe, reusable: an exhaustively verified toy action whose orbit, class number, and action-

group order agree at 9 and 27 (§3); a transparent simplified sieve with a finite-range constant $c = 2.68677\ [2.65454, 2.71923]$ and an explicit statement of everything that interval excludes (§4); a cost calculator whose every number is a named assumption and which reproduces the $2^{71.6}$ and 2^{56} published logical endpoints exactly (§5); a sensitivity ranking showing parallelism and oracle depth dominate the physical answer (§6); and a conditional $\Omega(\log N)$ query bound that is tight in its interface and provably silent outside it (§7). The obstruction analysis (§8) reframes Task 1: before asking for **the** constant, one must fix a named point in a four-dimensional assumption space — sieve schedule, oracle model, memory technology, error correction — and the literature has not converged on one. EMPIRICAL: final validation, 2026-06-30 All artifacts behind these statements passed the closing audit: 63 shared tests, 10 problem-specific tests, five script smoke modes, bytecode compilation, JSON parsing, and the principal CSV row and schema check.

Reproducibility

All results derive from the P3.2 research artifacts. The toy action is `code/verify_toy_action.py` (outputs `verify_toy_action_p59_20260630.json`, `verify_toy_action_p419_20260630.json`); the sieve pipeline is `code/simulate_sieve.py` and `code/fit_sieve.py` (outputs `simulate_sieve_n24-96_seed20260722_20260630.csv`, `fit_sieve_n24-96_20260630.json`, seeds 20260722/20260723); the cost layer is `code/cost_model.py` with the configurations `illustrative_surface_code.json`, `bonnetain_schrottenloher_2020_section3_3.json`, and `peikert_2020_optimistic_endpoint.json` (outputs the three `cost_model_*_20260630.json` reports), and `code/sensitivity.py` with `illustrative_sensitivity_ranges.json` (output `sensitivity_n64_20260630.json`). Every figure in this paper is generated from those files by `figures/P3.2/make.py`. Epistemic tags PROVED, CITED, EMPIRICAL: range, CONDITIONAL: interface, HEURISTIC are carried over verbatim from the research log; untagged sentences are exposition, not claims.

References

[1] W. Castryck, T. Lange, C. Martindale, L. Panny, and J. Renes, "CSIDH: An efficient post-quantum commutative group action," in *ASIACRYPT 2018*, 2018, pp. 395–427.

[2] G. Kuperberg, "A subexponential-time quantum algorithm for the dihedral hidden subgroup problem," *SIAM Journal on Computing*, vol. 35, no. 1, pp. 170–188, 2005.

[3] O. Regev, "A subexponential time algorithm for the dihedral hidden subgroup problem with polynomial space." 2004.

[4] G. Kuperberg, "Another subexponential-time quantum algorithm for the dihedral hidden subgroup problem," in *TQC 2013*, in LIPIcs, vol. 22. 2013, pp. 20–34.

[5] A. M. Childs, D. Jao, and V. Soukharev, "Constructing elliptic curve isogenies in quantum subexponential time," *Journal of Mathematical Cryptology*, vol. 8, no. 1, pp. 1–29, 2014.

[6] X. Bonnetain and A. Schrottenloher, "Quantum security analysis of CSIDH," in *EUROCRYPT 2020, Part II*, 2020, pp. 493–522.

[7] C. Peikert, "He gives C-sieves on the CSIDH," in *EUROCRYPT 2020, Part II*, 2020, pp. 463–492.

[8] D. J. Bernstein, T. Lange, C. Martindale, and L. Panny, "Quantum circuits for the CSIDH: Optimizing quantum evaluation of isogenies," in *EUROCRYPT 2019*, 2019, pp. 409–441.

[9] M. Ettinger, P. Høyer, and E. Knill, "The quantum query complexity of the hidden subgroup problem is polynomial," *Information Processing Letters*, vol. 91, no. 1, pp. 43–48, 2004.

[10] M. Remaud, A. Schrottenloher, and J.-P. Tillich, "Time and query complexity tradeoffs for the dihedral coset problem." 2022.

[11] D. Bacon, A. M. Childs, and W. van Dam, "Optimal measurements for the dihedral hidden subgroup problem," *Chicago Journal of Theoretical Computer Science*, vol. 2006, no. 2, 2006.