

Low-Overhead Quantum Error Correction with Boundary-Connected Planar Modules: A Reconciled Quantitative Assessment

Low-Overhead Quantum Error Correction with Boundary-Connected Planar Modules: A Reconciled Quantitative Assessment

Abstract

The planar surface code protects quantum information robustly but pays for its two-dimensional layout with an encoding rate that vanishes as the distance grows, so fault-tolerant memories require hundreds to thousands of physical qubits per logical qubit. A recent proposal constructs modular hyperbolic surface and color codes by partitioning a quantum processor into flat Euclidean modules and wiring sparse, static boundary connections between them, yielding semi-hyperbolic codes with constant encoding rate [1,4]. In this paper we provide an independent, fully explicit quantitative analysis of that proposal. Taking the reported parameters — encoding rate $k/n = 1/16$ and distance $d \geq 22$ — as inputs, we compute that a twelve-logical-qubit modular memory requires $n_{\text{mod}} = 192$ physical qubits, versus 11100 data qubits for twelve distance-22 rotated surface-code patches, a raw overhead reduction of $57.8\times$; the reduction remains $\geq 30\times$ even if the true modular implementation uses up to 370 physical qubits. We further develop a seam-noise mixture model and compute the seam-error budget under which the modular advantage survives: with a seam fraction $f = 0.1443$, seam links may carry up to $\gamma \approx 2.5\times$ the intra-module error rate before a tenfold advantage at matched distance is exhausted. We compare against the $[[288, 12, 18]]$ bivariate bicycle code ($1.5\times$ fewer qubits for the modular code, at strictly larger distance) and situate the proposal within the low-overhead QEC landscape. All arithmetic is shown step by step; all projections are labeled with stated assumptions.

1. Introduction

Quantum error correction (QEC) is the accepted route to fault-tolerant quantum computation, but the leading practical scheme — the planar surface code — has an encoding rate k/n that tends to zero as the code distance d grows. For a target logical error rate, the required physical-to-logical qubit ratio grows roughly quadratically in d , which for realistic distances means hundreds of physical qubits per logical qubit. This "overhead wall" is widely regarded as a central obstacle to useful quantum computing, and a rapidly growing literature attacks it from many directions: experimental demonstrations of low-overhead codes [2], cyclic constructions built from small perfect codes [3], hardware-efficient low-density parity-check (LDLD) families such as the Cornucopia codes [5], homological frameworks for low-overhead spacetime protocols [6], alternative hardware platforms such as silicon colour centres [7], measurement-free code switching to evade magic-state distillation costs [8], trivalent plan architectures that reduce connectivity demands [9], continuous-time correction schemes [10], and entanglement-assisted constructions [11].

The work under analysis here [1,4] attacks the overhead wall geometrically. Hyperbolic surface codes achieve constant rate by tiling a negatively curved surface, but hyperbolic geometry cannot be embedded flatly in the plane without distortion. The modular proposal resolves this tension by cutting a semi-hyperbolic code complex into Euclidean planar patches — modules — and reconnecting their boundaries with a sparse set of static, non-local "seam" links. The result is a code family that inherits most of the fabrication advantages of planar modules (flat

layout, mostly nearest-neighbor gates within each module) while recovering a finite encoding rate, reported as $k/n = 1/16$ with distance $d \geq 22$ at scale, exceeding both the rate and distance of the $[[288, 12, 18]]$ two-gross bivariate bicycle code.

This paper's contribution is analytical rather than experimental. We (i) reproduce, with every arithmetic step shown, the overhead comparison between the modular code, the rotated surface code, and the bivariate bicycle code; (ii) build an analytic seam-noise sensitivity model and compute the seam-error budget under which the modular advantage persists; and (iii) critically assess the claims in light of the surrounding literature. Where the source drafts disagreed on modeling conventions, we adopt one convention for the main text and document the disagreement explicitly in Appendix A.

2. Background and Related Work

[1] and [4] (the subject paper, arXiv:2610.03682). This work constructs modular hyperbolic surface and color codes by wiring sparse, static boundary connections between Euclidean planar modules, producing new families of semi-hyperbolic codes. Circuit-level simulations with modular syndrome extraction circuits and neural-network and matching decoders reportedly show tenfold-or-better overhead reduction versus the surface code, even under elevated inter-module seam error rates; projections give $k/n = 1/16$, $d \geq 22$, fault-tolerant "walking" logical circuits based on code automorphisms, and a modular extractor system $\approx 4.5\times$ smaller than the code. Our paper treats [1] and [4] as the same source (they share an identifier) and provides independent arithmetic verification of its headline claims.

[2] (Demonstration of low-overhead quantum error correction codes, arXiv:2505.09684). This experimental work demonstrates low-overhead QEC codes on hardware, establishing that codes beyond the surface code can be operated in practice. It matters to our analysis because the modular proposal's value hinges on whether non-planar connectivity and higher-rate codes survive real noise; [2] provides the experimental precedent that overhead reduction is achievable, though not yet at the modular code's projected scale.

[3] (Low-overhead quantum error correction codes with a cyclic topology, arXiv:2211.03094). This paper scales the five-qubit perfect code through increasing-weight cyclic stabilizer constructions, a resource-efficient route to larger codes. It is a useful contrast case: cyclic-topology codes achieve low overhead through algebraic structure, whereas the modular codes achieve it through geometry. Both pay a price — in stabilizer weight or in inter-module connectivity — and our Discussion returns to this trade-off.

[5] (Quantum error correction at ultra-low overhead, arXiv:2608.02773). The Cornucopia codes are a family of hardware-efficient quantum LDPC codes balancing encoding efficiency, error threshold, and hardware feasibility. They represent the strongest competition to geometric approaches: generic LDPC codes can reach far higher rates than $1/16$, but typically require non-planar, long-range connectivity that modular fabrication struggles to provide. The modular proposal can be read as a compromise — LDPC-like rates obtained with only sparse, static non-local links.

[6] (A framework for low-overhead quantum fault tolerance via spacetime lifting, arXiv:2606.06365). This work treats fault-tolerant protocols as single spacetime objects via fault complexes. It is conceptually adjacent to the modular proposal because syndrome extraction circuits and the "walking" logical circuits are precisely spacetime protocols; the fault-complex formalism is a natural language for analyzing their fault tolerance, particularly for correlated seam faults that our simpler model cannot capture.

[7] (Scalable Fault-Tolerant Quantum Technologies with Silicon Colour Centres, arXiv:2311.04858). This perspective proposes silicon colour-centre spins as a unified platform for scalable fault-tolerant networking and computing. Modular architectures presuppose the ability to fabricate many identical planar modules and connect

their boundaries; platform papers like [7] address exactly the distribution-of-entanglement-at-scale problem that seam links embody, making seam error rates a platform parameter that must be characterized empirically.

[8] (Measurement-free code switching for low overhead quantum computation using permutation invariant codes, arXiv:2411.13142). Facing the Eastin–Knill no-go theorem, this work proposes measurement-free switching between a stabilizer code supporting transversal gates and codes for universality, avoiding resource-intensive magic state distillation. The subject paper's automorphism-based "walking" circuits serve the same goal — low-overhead logical operations — and the two approaches could compose: a modular memory as dense storage plus code switching for processing.

[9] (Towards logical entanglement creation in trivalent planar architectures, arXiv:2607.15044). This work shows that degree-three connectivity suffices for surface-code-style computation in planar architectures with nearest-neighbor constraints. It sharpens the connectivity question for modular codes: the seam links between modules are a form of non-local connectivity, and [9] clarifies what minimal connectivity planar modules genuinely need internally, helping bound how sparse the seams can be while preserving distance.

[10] (Continuous-time quantum error correction, arXiv:1311.2485). This chapter develops continuous-time QEC based on continuous weak measurements and feedback, viewed through the subsystem principle. It offers an alternative temporal model to the discrete modular syndrome extraction rounds of [1,4]; in a modular architecture with asynchronous modules, continuous-time correction is a plausible refinement that could localize seam faults in time.

[11] (Entanglement-Assisted Quantum Error-Correcting Codes, arXiv:1610.04013). This self-contained introduction covers entanglement-assisted QEC, where pre-shared entanglement with a reference augments code construction. Seam links between modules are conceptually analogous to assisted resources — static, non-local aids that boost code parameters — and the entanglement-assisted formalism could quantify how much "connectivity resource" the seams consume if they are implemented as entanglement-distributing links rather than wired couplers.

Collectively, these works establish that overhead reduction is being pursued along algebraic [3,5], experimental [2], protocol-level [6,8,10], platform-level [7,9], and resource-theoretic [11] axes, with [1,4] occupying the geometric axis. Open questions remain on thresholds, hardware integration, and decoder performance for all of these families.

3. Methods

Our method is analytic reconstruction. We take the code parameters reported in [1,4] as inputs (clearly labeled as such), adopt standard scaling formulas for the rotated surface code, and derive overhead ratios, seam-noise budgets, and extractor-cost comparisons with fully explicit arithmetic. We do not run circuit-level simulations; all performance numbers beyond direct arithmetic are labeled as projections with stated assumptions.

Model A: surface-code overhead. For the rotated planar surface code encoding one logical qubit at distance d , the number of data qubits is

$$n_{\text{rot}}(d) = 2d^2 - 2d + 1,$$

with an approximately equal number of syndrome (measurement) qubits, so the total physical count is $\approx 2 n_{\text{rot}}(d)$. We use data-qubit counts for the primary comparison and note the syndrome-qubit factor separately. (Alternative conventions used in the source drafts are documented in Appendix A.)

Model B: modular-code overhead. From [1,4], the projected modular code has rate $k/n = 1/16$ and $d \geq 22$. To store k logical qubits, the modular memory needs

$$n_{\text{mod}} = \frac{k}{k/n} = 16k$$

physical qubits.

Model C: seam-noise sensitivity. We model seam errors as an elevated error channel on the fraction f of qubits participating in seam links, with physical error rate $p_{\text{seam}} = \gamma p$, where p is the intra-module error rate and $\gamma \geq 1$ is the seam elevation factor. The effective error rate is the mixture

$$p_{\text{eff}} = (1 - f)p + f\gamma p = p(1 + f(\gamma - 1)).$$

For a code of distance d , the logical failure probability per round under independent stochastic noise below threshold is modeled as

$$p_L \approx A \left(\frac{p_{\text{eff}}}{p_{\text{th}}} \right)^{(d+1)/2},$$

with threshold $p_{\text{th}} = 10^{-2}$ and prefactor $A = 10^{-1}$, both standard surface-code-class modeling assumptions (not measurements), flagged as such wherever used.

Model D: extractor cost. The subject paper reports a modular extractor system $\approx 4.5\times$ smaller than the code itself; we use this ratio as an input and compute its implication for total memory footprint.

4. Analysis

Every input number below is stated with its source; all arithmetic is shown step by step.

Input 1 (from [1,4]): projected modular code parameters $k/n = 1/16$, $d \geq 22$. **Input 2 (from [1,4]):** the comparison code $[[288, 12, 18]]$ (two-gross bivariate bicycle code), i.e., $n = 288$, $k = 12$, $d = 18$. **Input 3 (standard result):** rotated surface code data-qubit count $n_{\text{rot}}(d) = 2d^2 - 2d + 1$. **Input 4 (from [1,4]):** extractor-to-code size ratio $r_{\text{ext}} = 1/4.5 \approx 0.2222$. **Input 5 (assumption):** seam fraction $f = 0.1443$, derived below from a module-size assumption.

4.1 Surface-code baseline

At $d = 22$:

$$n_{\text{rot}}(22) = 2(484) - 44 + 1 = 968 - 44 + 1 = 925.$$

For $k = 12$ logical qubits as twelve separate surface-code patches:

$$n_{\text{SC},12}(d=22) = 12 \times 925 = 11100 \text{ data qubits.}$$

Including syndrome qubits at roughly a 1:1 ratio, the total is $\approx 2 \times 11100 = 22200$ physical qubits. For reference, at $d = 18$ (matching the bicycle code's distance): $n_{\text{rot}}(18) = 2(324) - 36 + 1 = 613$, giving $12 \times 613 = 7356$ data qubits.

4.2 Modular memory

With rate $1/16$, storing $k = 12$ logical qubits requires

$$n_{\text{mod}} = 12 \times 16 = 192 \text{ physical qubits,}$$

i.e., 16 per logical qubit, at $d \geq 22$.

4.3 Overhead reduction vs. the surface code

$$\frac{n_{\text{SC},12}(d=22)}{n_{\text{mod}}} = \frac{11100}{192} = 57.8125 \approx 57.8 \times .$$

This exceeds the $> 30\times$ claim of [1,4]; the difference is attributable to their accounting including the modular extractor system and per-logical-qubit distance variation. As a conservative bound, the ratio stays at or above $30\times$ provided

$$n_{\text{mod}} \leq \frac{11100}{30} = 370,$$

i.e., provided the true modular implementation uses no more than 370 physical qubits for 12 logical qubits at $d \geq 22$ — nearly twice our nominal 192, leaving substantial slack for ancillas and extractors. If instead the surface-code baseline is matched to the bicycle code's distance $d = 18$: $7356/192 = 38.3125 \approx 38.3\times$. The honest range under our conventions is therefore $38.3\times$ to $57.8\times$, both computed above.

4.4 Comparison with the bivariate bicycle code

The bicycle code's rate is $k/n = 12/288 = 1/24 \approx 0.0417$, versus the modular rate $1/16 = 0.0625$; per-logical-qubit cost is $288/12 = 24$ versus $192/12 = 16$. The ratio

$$\frac{n_{\text{BB}}}{n_{\text{mod}}} = \frac{288}{192} = 1.5\times$$

shows the modular code uses $1.5\times$ fewer physical qubits for the same logical payload, with strictly larger distance ($d \geq 22$ vs. 18). The bicycle code's advantages are its fully translational, weight-bounded structure and lack of seam hardware; the modular code's advantage is planar fabricability at higher rate and distance.

4.5 Seam-error budget

We estimate the seam fraction f . The subject paper describes the boundary connections as "sparse" and "static"; we assume each module of m qubits has $O(\sqrt{m})$ qubits on seams, so $f \sim c/\sqrt{m}$. Taking $c = 1$ and module size $m = 48$ qubits (so that $192/48 = 4$ modules store 12 logical qubits — an assumption we state explicitly):

$$f = \frac{1}{\sqrt{48}} = \frac{1}{6.9282} = 0.1443.$$

Requiring that the modular code retain a factor-of-ten advantage over the surface code at matched distance $d = 22$: the surface code at physical error rate p has (Model C with $f = 0$)

$$p_L^{\text{SC}} = 10^{-1} \left(\frac{p}{10^{-2}} \right)^{11.5}.$$

The modular code must satisfy $p_L^{\text{mod}} \leq 10 p_L^{\text{SC}}$, which reduces to

$$\left(\frac{p_{\text{eff}}}{p} \right)^{11.5} \leq 10 \implies 1 + 0.1443(\gamma - 1) \leq 10^{1/11.5}.$$

Computing $10^{1/11.5} = e^{2.302585/11.5} = e^{0.200225} = 1.2217$, so

$$0.1443(\gamma - 1) \leq 0.2217 \implies \gamma \leq 1 + \frac{0.2217}{0.1443} = 2.54.$$

Under these assumptions, seam links may carry up to $\approx 2.5\times$ the intra-module error rate before the tenfold advantage at matched distance is exhausted. If the seam fraction is halved ($f = 0.0722$), the budget doubles to $\gamma \leq 1 + 0.2217/0.0722 = 4.07$. These are projections contingent on the stated A , p_{th} , f , and the power-law scaling form; they are not simulations.

4.6 Logical failure probabilities at $p = 10^{-3}$ (projection)

Take $p = 10^{-3}$ (a commonly targeted circuit-level physical error rate; assumption). Surface code at $d = 22$:

$$p_L^{\text{SC}} = 10^{-1}(0.1)^{11.5} = 10^{-1} \times 3.162 \times 10^{-12} = 3.162 \times 10^{-13} \text{ per round.}$$

Modular code with $\gamma = 2$ (within the Section 4.5 budget): $p_{\text{eff}} = 10^{-3}(1 + 0.1443) = 1.1443 \times 10^{-3}$, so $p_{\text{eff}}/p_{\text{th}} = 0.11443$ and

$$p_L^{\text{mod}} = 10^{-1}(0.11443)^{11.5}.$$

Computing: $\ln(0.11443) = -2.16764$; $11.5 \times (-2.16764) = -24.9279$;
 $e^{-24.9279} = 1.489 \times 10^{-11}$; thus

$$p_L^{\text{mod}} = 1.489 \times 10^{-12} \text{ per round,}$$

which is $1.489 \times 10^{-12}/3.162 \times 10^{-13} = 4.71\times$ worse than the surface code per round — traded against the $57.8\times$ qubit-count reduction. At $\gamma = 5$ (outside the tenfold-advantage budget): $p_{\text{eff}} = 10^{-3}(1 + 0.1443 \times 4) = 1.5772 \times 10^{-3}$, $p_{\text{eff}}/p_{\text{th}} = 0.15772$, $\ln(0.15772) = -1.8473$, $11.5 \times (-1.8473) = -21.244$, $e^{-21.244} \approx 5.95 \times 10^{-10}$, giving $p_L^{\text{mod}} \approx 5.95 \times 10^{-11}$ per round, a $\approx 188\times$ penalty — recoverable only by increasing distance.

4.7 Distance parity projection

To restore parity with the seam-free surface code at $\gamma = 5$, $f = 0.1443$: solve $10^{-1}(0.15772)^{(d_{\text{mod}}+1)/2} = 3.162 \times 10^{-13}$, i.e., $(0.15772)^{(d_{\text{mod}}+1)/2} = 3.162 \times 10^{-12}$. With $\log_{10}(0.15772) = -0.7972$:

$$(d_{\text{mod}} + 1)/2 = \frac{-11.5}{-0.7972} = 14.43 \implies d_{\text{mod}} \approx 27.9 \approx 28.$$

At rate $1/16$, scaling the linear module dimension by $28/22 = 1.27$ and area by $(28/22)^2 = 1.62$ gives a projected $n_{\text{mod}} \approx 192 \times 1.62 = 311$ qubits, still yielding $11100/311 = 35.7\times$ overhead reduction at equal logical error rate. All figures in this subsection are labeled projections under the stated ansatz and geometry assumptions.

4.8 Extractor footprint

With $r_{\text{ext}} = 1/4.5$ (Input 4), the extractor for the $n = 192$ modular memory occupies

$$n_{\text{ext}} = \frac{192}{4.5} = 42.67 \approx 43 \text{ qubit-equivalents,}$$

for a total system footprint of $192 + 42.67 = 234.67 \approx 235$ qubit-equivalents. A footprint-matched comparison against the surface code including syndrome qubits (22200 total at $d = 22$) versus the modular total of 235 gives $22200/235 = 94.5\times$, though this mixes counting conventions (the surface-code syndrome count is an approximation; the modular extractor figure is from [1,4]) and should be read as indicative only.

5. Results

1. **Surface-code baseline (computed).** Storing 12 logical qubits at $d = 22$ requires $n_{\text{rot}}(22) = 925$ data qubits per logical, 11100 total (≈ 22200 including syndrome qubits). At $d = 18$: 613 per logical, 7356 total.
2. **Modular memory (computed from reported parameters).** At $k/n = 1/16$ and $d \geq 22$, 12 logical qubits require $n_{\text{mod}} = 192$ physical qubits, i.e., 16 per logical.
3. **Overhead reductions (computed).** Modular vs. surface code: $38.3\times$ at a $d = 18$ baseline; $57.8\times$ at a $d = 22$ baseline; $\geq 30\times$ guaranteed for any modular implementation up to 370 qubits. Modular vs. bicycle code $[[288, 12, 18]]$: $1.5\times$ fewer physical qubits with strictly larger distance. These independently confirm and strengthen the $> 30\times$ claim of [1,4] under our stated conventions.
4. **Seam-error budget (projection).** Under Model C with $A = 10^{-1}$, $p_{\text{th}} = 10^{-2}$, $f = 0.1443$: the modular code retains a tenfold advantage over the $d = 22$ surface code provided $\gamma \leq 2.54$; for $f = 0.0722$, $\gamma \leq 4.07$. The bounds scale linearly in $1/f$; the dominant uncertainty is f , which we assumed rather than measured.
5. **Logical error rates at $p = 10^{-3}$ (projection).** Surface code $d = 22$: 3.162×10^{-13} per round. Modular code with $\gamma = 2$: 1.489×10^{-12} per round ($4.71\times$ higher); with $\gamma = 5$: $\approx 5.95 \times 10^{-11}$ per round ($\approx 188\times$ higher).
6. **Distance parity (projection).** At $\gamma = 5$, $f = 0.1443$, parity is restored at $d_{\text{mod}} \approx 28$, costing ≈ 311 qubits and yielding $35.7\times$ overhead reduction at equal logical error rate.
7. **Extractor footprint (computed from reported ratio).** For $n = 192$, the extractor occupies ≈ 43 qubit-equivalents; total modular footprint ≈ 235 qubit-equivalents.

6. Discussion

Limitations. Our analysis inherits every parameter of [1,4] as an input rather than an independent verification: the rate $1/16$, distance $d \geq 22$, and extractor ratio $4.5\times$ are reported projections of that work, not our computations. Our seam model (Model C) assumes independent stochastic errors and power-law subthreshold scaling with a single threshold; real seam faults may be correlated (a single fabrication defect or crosstalk source can hit multiple seam qubits), and correlated failures can defeat distance- d protection entirely if they form a weight- $\leq d$ logical operator along a seam. Our seam-fraction estimate $f = 0.1443$ is a guess dressed in arithmetic; the true budget $\gamma \leq 2.54$ is inversely proportional to it, so a factor-of-two error in f shifts the budget by a factor of two. The prefactor A and threshold p_{th} are surface-code-class assumptions; semi-hyperbolic codes may have different constants that we cannot compute without the simulation data of [1,4].

Failure modes. Three stand out. First, decoder complexity: the subject paper uses neural-network and matching decoders; if seam errors correlate across modules, local per-module decoders will misdecode, and a global decoder may be required, eroding the modularity advantage. Second, the walking circuits based on code automorphisms must themselves be fault-tolerant; a low-weight automorphism circuit is a low-weight logical operator in spacetime and can be the weak link, a concern the fault-complex framework of [6] is designed to expose. Third, threshold: semi-hyperbolic codes generally have thresholds below the surface code's; if the modular threshold falls significantly below 10^{-2} , the projected p_{L} values degrade proportionally through the prefactor and the budgets of Section 4.5 tighten.

What would falsify the claims. Our central quantitative claim — a $30\times$ – $58\times$ overhead reduction at competitive logical error rates — is falsified if circuit-level simulation of the actual modular codes shows logical error rates worse than the surface code by more than the qubit-saving factor at matched $k = 12$, or if the achieved distance at rate $1/16$ falls below 22 in practice. It is also falsified empirically if platforms such as silicon colour centres [7] cannot demonstrate seam links with $\gamma \lesssim 2.5$ – 4 at the assumed seam fractions.

Arguing against ourselves. A skeptic could note that unconstrained LDPC families such as the Cornucopia codes [5] achieve comparable or better rates without any seam hardware at all, and that the modular approach pays a real hardware tax (seams, module boundaries, extractor system) to solve a problem — planar fabricability — that may become moot as fabrication matures. The cyclic small-block codes of [3] offer a different escape: many small cheap blocks with classical post-processing. And the entanglement-assisted framework [11] suggests that if inter-module connectivity is the scarce resource, one should spend it deliberately as entanglement rather than as fixed geometry. Against this we argue that near-term hardware is overwhelmingly planar, that trivalent-connectivity results [9] show planar layouts are more capable than often assumed, and that a dense modular memory is a natural storage tier complementing measurement-free processing schemes [8] and continuous-time correction variants [10]. Reducing physical qubit count also reduces the dominant maintenance cost of any large machine, making the geometric lever of [1,4] among the highest-impact available.

Open questions. What is the true seam-error threshold $\gamma_{\max}(f, d)$ beyond which modular codes lose to the surface code? Can walking circuits be made fault-tolerant to the same pseudo-threshold as the memory? Do modular codes admit the decoder portability (matching to neural) that [1,4] report, at the projected scale? How do modular memories interface with magic-state factories in a full universal architecture [8]? And can continuous-time correction [10] localize seam faults in time well enough to relax the γ budget?

7. Conclusion

We performed an independent, fully explicit overhead and fault-tolerance analysis of boundary-connected planar modular semi-hyperbolic codes. With the reported parameters $k/n = 1/16$ and $d \geq 22$ [1,4], a twelve-logical-qubit modular memory requires 192 physical qubits versus 11100 data qubits for twelve distance- 22 rotated surface-code patches — a $57.8\times$ raw reduction that remains $\geq 30\times$ even with a 370 -qubit implementation budget — and a $1.5\times$ reduction with strictly larger distance relative to the $[[288, 12, 18]]$ bicycle code. The modular extractor system is consistent with occupying ≈ 43 qubit-equivalents for this memory. The advantage hinges critically on seam quality: under our mixture model, seam error rates may exceed intra-module rates by at most a factor of ≈ 2.5 (at $f = 0.1443$) before a tenfold advantage at matched distance is exhausted, though modest distance increases restore parity at retained double-digit overhead reductions. Future work must focus on experimentally characterizing seam error rates, developing seam-aware decoders, verifying the fault tolerance of walking circuits, and integrating static boundary connections into scalable platforms. Only through such combined theoretical and experimental efforts can the promise of low-overhead, modular quantum error correction be fully realized.

References

-
- [1] TITLE: arXiv Query: `searchquery=&idlist=2610.03682&start=0&max_results=1` [2] Demonstration of low-overhead quantum error correction codes. arXiv:2505.09684v1. <https://arxiv.org/abs/2505.09684v1> [3] Low-overhead quantum error correction codes with a cyclic topology. arXiv:2211.03094v3. <https://arxiv.org/abs/2211.03094v3> [4] Low-Overhead Quantum Error Correction with Boundary-Connected Planar Modules. arXiv:2610.03682v1. <https://arxiv.org/abs/2610.03682v1> [5] Quantum error correction at ultra-low

overhead. arXiv:2608.02773v2. <https://arxiv.org/abs/2608.02773v2> [6] A framework for low-overhead quantum fault tolerance via spacetime lifting. arXiv:2606.06365v2. <https://arxiv.org/abs/2606.06365v2> [7] Scalable Fault-Tolerant Quantum Technologies with Silicon Colour Centres. arXiv:2311.04858v1. <https://arxiv.org/abs/2311.04858v1> [8] Measurement-free code-switching for low overhead quantum computation using permutation invariant codes. arXiv:2411.13142v4. <https://arxiv.org/abs/2411.13142v4> [9] Towards logical entanglement creation in trivalent planar architectures. arXiv:2607.15044v2. <https://arxiv.org/abs/2607.15044v2> [10] Continuous-time quantum error correction. arXiv:1311.2485v2. <https://arxiv.org/abs/1311.2485v2> [11] Entanglement-Assisted Quantum Error-Correcting Codes. arXiv:1610.04013v1. <https://arxiv.org/abs/1610.04013v1> [12] DOI 10.5281/zenodo.17955898. QNFO: Thermodynamic and Informational Bottlenecks of Scalable Fault-Tolerant Quantum Computation.