

Code-Agnostic Graph Neural Network Decoding from Detection Error Models: A Reconciled Quantitative and Structural Assessment

Code-Agnostic Graph Neural Network Decoding from Detection Error Models: A Reconciled Quantitative and Structural Assessment

Abstract

Graph neural network (GNN) decoders for quantum error correction have historically been locked to specific code families. The POLYMECHANON preprint [1,2] proposes a decoder whose sole input is the detection error model (DEM) — a tripartite graph of detectors, error mechanisms, and logical observables — making the code data rather than an architectural design choice. This paper provides an independent, arithmetic-first assessment of that claim. We convert the reported headline numbers into concrete quantities: a 25% reduction in logical failures versus correlated MWPM on the rotated surface code corresponds to an improvement factor of 0.75, which under the standard scaling ansatz $pL \propto (p/p_{th})^{d/2}$ is equivalent to a relative pseudo-threshold improvement of $\approx 10.06\%$ at $d = 5$ and $\approx 7.46\%$ at $d = 7$; a 16% reduction on the $[[130,4,6]]$ qLDPC code corresponds to a factor of 0.84 and an equivalent threshold improvement of $\approx 5.98\%$. The reported latency advantage ("a few ms per shot on a single GPU" versus "a few tens of ms" for BP+OSD on a CPU core) would bound the speedup in $[4\times, 25\times]$, midpoint $\approx 10\times$, with an explicit hardware-asymmetry caveat — but this latency analysis is conditional on claims not verifiable in the available abstract record of [1,2]. We derive DEM graph sizes for canonical cases (441 detectors, 1,337 mechanisms, 2,681 edges at $d = 8$, $R = 8$ under stated conventions), prove a cheap non-isomorphism certificate between surface-code and qLDPC DEMs via part-size vectors, and show that the claimed post-selection result ($>10\times$ error reduction at $>90\%$ shot retention) would arithmetically require discarded shots to fail at $\approx 9.1\times$ the base rate — a strong, testable separability condition, conditional on a claim not verifiable in the available abstract record of [1,2]. We identify size-generalization, DEM fidelity, and hardware-fair latency comparison as the principal open risks, and propose falsification experiments for each.

1. Introduction

Fault-tolerant quantum computation requires a classical decoding layer that converts syndrome measurement streams into logical corrections faster than errors accumulate. The dominant decoders — minimum-weight perfect matching (MWPM) for surface-type codes and belief propagation with ordered-statistics post-processing (BP+OSD) for quantum low-density parity-check (qLDPC) codes — are hand-engineered and either accuracy-limited (MWPM ignores error correlations) or latency-limited (BP+OSD runtime grows with physical error rate and code size).

Neural decoders promise learned inference at constant time, but most published architectures hard-code the code family: the input representation (e.g., a $d \times d \times R$ syndrome tensor) encodes the geometry. POLYMECHANON [1,2] proposes a different contract: the decoder consumes only the DEM, a code-agnostic object any stabiliser code under any expressible noise model can emit, rendered as a tripartite graph with all input features computed from the code as data.

This paper offers a reconciled independent analysis with three contributions: (i) explicit derivations converting the reported relative improvements into threshold shifts, distance equivalents, and throughput budgets, with every input number sourced to [1,2] and all assumptions labeled; (ii) a structural analysis of the DEM-as-graph representation, including size scaling and isomorphism-invariance bounds on what any DEM-only decoder can achieve; and (iii) a

structured critique with falsifiable predictions. Where our source drafts disagreed on conventions (representative latencies, baseline failure rates, distance points), we adopt one convention in the main text and document every conflict in Appendix A.

2. Background and Related Work

The decoder under analysis [1,2]. POLYMECHANON is a GNN decoder whose sole input is the DEM, represented as a tripartite graph of detectors, error mechanisms, and logical observables. Its headline claims: up to 25% fewer logical failures than correlated MWPM on the rotated surface code under phenomenological and circuit-level noise; parity with BP+OSD on small high-rate qLDPC codes and up to 16% fewer failures on $[[130,4,6]]$; physical-error-rate-independent decoding time (a few ms per shot on one GPU versus tens of ms for BP+OSD on one CPU core); a cross-code model that beats uncorrelated MWPM on seen graph-like codes, stays within 10% of BP+OSD elsewhere, but fails to generalize to unseen larger codes; and confidence-based post-selection lowering the logical error rate by more than an order of magnitude while retaining >90% of shots. These are the sole empirical inputs to our derivations.

Learned message passing for classical codes [5]. This work establishes that a fully differentiable GNN can learn a generalized message-passing algorithm over the graph of a forward-error-correction code, achieving competitive performance on LDPC and BCH codes. It is the direct classical ancestor of DEM-based quantum decoding: both recast decoding as inference on a code-defined graph. The key difference is that the Tanner graph is fixed per code, whereas the DEM graph changes with both code and noise model — raising the transfer question POLYMECHANON must answer.

Universal graph embeddings via transfer learning [4]. This work pretrains GNN embeddings on large corpora with transfer-learning aids, motivated by embedding successes in language and vision, and documents that existing GNN embeddings do not transfer robustly across graph distributions without deliberate design. POLYMECHANON's cross-code model is a concrete instance of this program, and its reported failure on unseen larger codes is consistent with [4]'s central observation — suggesting the generalization gap is a known structural difficulty of graph transfer learning, not an implementation bug.

Sparse defect-centric decoding [7]. The Sparse Mamba Decoder attacks the cost side of neural QEC decoding: most neural decoders process the full dense syndrome array of size $O(d^2R)$ regardless of the actual error rate, whereas defect-centric sparse processing scales with detected defects. POLYMECHANON's constant-time claim is complementary but different: [7] makes decoding cheaper at low error rates by exploiting sparsity; POLYMECHANON makes it constant by using a fixed computational graph whose size is set by the code, not the realized error pattern. Their interaction is an open question we quantify structurally in Section 4.

Adaptive-topology graph architectures [6]. GraphFPN shows that multi-scale feature learning benefits when network topology adapts to the input rather than remaining fixed. This is salient because a DEM decoder must handle graphs of wildly varying size and structure (geometrically local surface-code DEMs versus long-range qLDPC DEMs); whether fixed-depth message passing suffices for long-range qLDPC correlations is a question GraphFPN's findings make pressing, and scale-pyramidal variants are a natural but untested remedy for size-generalization failure.

Random neural networks [3]. This tutorial frames neural computation as queuing-network analysis over 25 years of applications. The view is useful here because DEM decoding is fundamentally a probabilistic flow problem: error mechanisms inject probability mass that must be routed to detectors. It is also a reminder that "neural decoder" is not new; what is new in [1,2] is the input contract (the DEM), not the neural machinery.

Transfer learning for recurrent predictors [8]. The LSTM air-pollutant transfer study exemplifies the standard pretrain-then-finetune recipe and its standard failure mode — degradation on target domains far from the source. It serves as a calibration point for POLYMECHANON's cross-code claims and suggests the pretrain-then-finetune alternative deserves direct comparison against joint multi-code training.

Semi-supervised graph anomaly detection [9]. Mul-GAD aggregates multi-view information for anomaly detection and notes challenges GNN methods face relative to shallow baselines. Syndrome decoding is formally an anomaly-localization task on the DEM graph (defects are anomalous detector firings), and the tripartite DEM is precisely a multi-view heterogeneous graph, so its aggregation lessons bear directly on how detector, mechanism, and observable messages should be combined.

QNFO frameworks [10,11,12,13]. The QNFO network-isomorphism framework [10] supplies the formal machinery — graph invariants, isomorphism testing — that we adapt to DEM graphs: two codes whose DEMs are isomorphic as attributed graphs are indistinguishable inputs to any code-agnostic decoder, bounding what DEM-only decoding can achieve. Topological quantization and spectral filtration [11] supply spectral invariants usable as cheap non-isomorphism certificates. The ultrametric-tree attention proof-of-concept [13] suggests hierarchical distance-based attention as an alternative to flat message passing, relevant to long-range detector correlations. Finally, the QNFO audit of the BQNN quantum neural network [12] exemplifies the due-diligence standard — staged external verification of quantum-ML claims — that this paper applies in miniature, restricting quantitative claims to what the source states and labeling all extrapolations as projections.

3. Methods

Definitions. A stabiliser code encodes k logical qubits into n physical qubits with distance d . Repeated syndrome measurement produces *detectors*: parity checks that must be 0 in the absence of errors. The *detection error model* (DEM) lists error *mechanisms*, each with probability p_i and a fault signature — the detectors it flips and the logical observables it anticommutes with. Decoding: given detector firings, infer the most likely logical observable flips.

Analytical pipeline.

1. **Claim extraction.** From [1,2] we extract: (C1) up to 25% fewer logical failures than correlated MWPM on the rotated surface code; (C2) up to 16% fewer than BP+OSD on $[[130,4,6]]$; (C3) decoding time independent of physical error rate, a few ms/shot on one GPU versus tens of ms for BP+OSD on one CPU core; (C4) cross-code model within 10% of BP+OSD on seen families, failing on unseen larger codes; (C5) post-selection: $>10\times$ logical-error reduction at $>90\%$ shot retention.
1. **Unit normalization.** "X% fewer logical failures" is a relative reduction in p_L at fixed physical error rate and code. Improvement factor $F = 1 - X/100$: $F_1 = 0.75$, $F_2 = 0.84$.
1. **Model-based translation.** We use the standard sub-threshold scaling ansatz $p_L(p, d) = A \cdot (p/p_{th})^{\lceil d/2 \rceil}$ as a translation device only, stating its assumptions wherever applied. It is well supported for surface codes, only heuristic for general qLDPC codes.
1. **Structural sizing.** We model the DEM as an attributed tripartite graph $G = (D \cup M \cup L, E)$ and derive node/edge counts for canonical cases under explicitly stated construction conventions.
1. **Throughput budgeting.** Latency claims are bounded using the published ranges at face value, with the GPU-vs-CPU hardware asymmetry stated as a caveat; real-time feasibility is assessed against a parameterized cycle time T_{cycle} , labeled as a projection.

No numbers are invented; every derived quantity traces to [1,2] plus labeled assumptions. However, the available source record for [1,2] is a truncated abstract containing only the 25% and 16% failure-reduction figures; the latency inputs (C3), the cross-code performance claims (C4), and the post-selection claims (C5) cannot be verified against it. All analyses in §4.3, §4.4, and §4.7 — including the speedup bounds [4×, 25×], the 9.1× separability requirement, and the [1.0, 1.31] penalty bracket — are therefore conditional projections contingent on claims not verifiable in the available abstract record, and should be checked against the full POLYMECHANON preprint (its latency, post-selection, and cross-code evaluation sections/tables) before use.

4. Analysis

4.1 Improvement factor and effective threshold shift (C1)

Input [1,2]: up to 25% fewer logical failures than correlated MWPM. The relative logical error rate is $pL^{POLY} / pL^{MWPM} = 0.75$. Under the ansatz with $m = \lceil d/2 \rceil$, holding the operating point fixed:

$$(pth^{MWPM} / pth^{POLY})^m = 0.75 \implies pth^{POLY} / pth^{MWPM} = (1/0.75)^{(1/m)} = (4/3)^{(1/m)}.$$

At $d = 5$ ($m = 3$): $\ln(4/3) = 1.386294 - 1.098612 = 0.287682$; $0.287682/3 = 0.095894$; $e^{0.095894} = 1.100642416298209$, i.e., a $\approx 10.06\%$ relative threshold improvement. At $d = 7$ ($m = 4$): $e^{(0.287682/4)} = e^{0.0719206} = 1.074569931823542$, i.e., $\approx 7.46\%$. At $d = 9$ ($m = 5$): $e^{0.0575364} = 1.0592238410488122$, i.e., $\approx 5.92\%$. The same relative improvement corresponds to a larger threshold shift at smaller distance — a caveat for extrapolation.

Distance-equivalent reading. At fixed threshold with $r = p/p_{th} = 0.5$, the exponent gain is $\Delta m = \ln 0.75 / \ln 0.5 = (-0.287682)/(-0.693147) = 0.4150374992788438$. One odd-distance step ($\Delta d = 2$) gives a factor 0.5 at $r = 0.5$; the 25% improvement is thus roughly half of one distance step at that operating point.

4.2 qLDPC improvement (C2)

Input [1,2]: up to 16% fewer logical failures than BP+OSD on $[[130,4,6]]$ ($n = 130$, $k = 4$, $d = 6$; encoding rate $k/n = 4/130 = 0.03076923076923077 \approx 3.08\%$). Improvement factor 0.84; $m = \lceil 6/2 \rceil = 3$:

$$pth^{POLY} / pth^{BP+OSD} = (1/0.84)^{(1/3)}; 1/0.84 = 1.190476; \ln = 0.174353; /3 = 0.0581177; e^{0.0581177} = 1.0598398329483265.$$

So $\approx 5.98\%$ relative threshold improvement, illustrative only given the weaker standing of the ansatz for qLDPC codes.

4.3 Latency and throughput (C3)

Inputs [1,2]: "a few ms" per shot on one GPU; "a few tens of ms" for BP+OSD on one CPU core, on $[[130,4,6]]$.

Conditional: these latency figures are not verifiable in the available abstract record of [1,2]; the following is a conditional projection contingent on the full preprint's latency section. Taking the stated ranges at face value: $t_G \in [2, 5]$ ms, $t_B \in [20, 50]$ ms. Speedup bounds: lower $20/5 = 4\times$; upper $50/2 = 25\times$; midpoint $35/3.5 = 10\times$. Throughputs at midpoints: $1/0.0035$ s ≈ 286 shots/s/GPU versus $1/0.035$ s ≈ 29 shots/s/core; matching one GPU would require ≈ 10 CPU cores, ignoring BP+OSD's error-rate-dependent slowdown.

Real-time projection (labeled). For a neutral-atom-style cycle budget $T_{cycle} = 10$ ms (a parameter choice, not a measurement from [1,2]), POLYMECHANON at 3 ms leaves a 7 ms margin (range 5–8 ms across the "few ms" ambiguity); BP+OSD at 30 ms exceeds the budget by 20 ms. The conclusion "candidate for real-time decoding" holds if $T_{cycle} \geq 3$ ms and if GPU-to-control-system transfer adds negligible latency — a cost the abstract does not discuss.

4.4 Post-selection consistency (C5)

Inputs [1,2]: error rate lowered by more than an order of magnitude while keeping >90% of shots. **Conditional:** this post-selection claim is not verifiable in the available abstract record of [1,2]; the derivation below is a conditional projection contingent on the full preprint's post-selection analysis. Let the retained fraction be $f \geq 0.9$, the post-selected rate $pL' = pL/10$ (conservative reading), and the discarded shots' rate p_L^{disc} . Conservation of failures:

$$pL = fpL' + (1-f)p_L^{\text{disc}}. \text{ With } f = 0.9: 0.1 \cdot p_L^{\text{disc}} = pL - 0.09 \cdot pL = 0.91 \cdot pL \implies p_L^{\text{disc}} = 9.1 \cdot p_L.$$

The discarded 10% must therefore carry failures at $\approx 9.1 \times$ the base rate — a strong separability requirement on the learned confidence. Testable: if the discarded rate were only $2 \cdot pL$, the arithmetic $(0.9 \cdot 0.1 + 0.1 \cdot 2) \cdot pL = 0.29 \cdot pL \neq pL$ would contradict the claim. **Throughput economics:** acceptance $a = 0.9$ costs a discarded fraction $0.1/0.9 \approx 11.1\%$ of throughput for a $\geq 10 \times$ error reduction; the accuracy-per-unit-throughput figure of merit improves by $r/a \geq 10/0.9 \approx 11.1 \times$.

4.5 DEM graph sizing (structural derivation)

Surface code, phenomenological noise. The rotated surface code has $d^2 - 1$ stabilisers ($(d^2-1)/2$ per basis). With the common convention that the first round yields no detectors, $ND = (d^2 - 1)(R - 1)$. At $d = 8$, $R = 8$: $d^2 - 1 = 63$; $R - 1 = 7$; $ND = 63 \times 7 = \mathbf{441 \text{ detectors}}$. Mechanisms: $n = d^2 = 64$ data qubits, two error types $\implies 128$ data-error mechanisms per round, each flipping 2 detectors; plus 63 measurement-error mechanisms per boundary, each flipping 1 detector. Over 7 boundaries: $N_M = (128 + 63) \times 7 = 191 \times 7 = \mathbf{1,337 \text{ mechanisms}}$. Edges into D: $128 \times 7 \times 2 + 63 \times 7 \times 1 = 1,792 + 441 = \mathbf{2,233}$. Edges into L ($k = 1$, Z-type mechanisms only): $64 \times 7 = \mathbf{448}$. Totals: $441 + 1,337 + 1 = \mathbf{1,779 \text{ nodes}}$; $2,233 + 448 = \mathbf{2,681 \text{ edges}}$. Scaling: $O(d^2R)$ in all counts — confirming the density concern of [7]: graph size is fixed by code and noise model, independent of realized error rate.

[[130,4,6]] qLDPC (projection, assumptions stated). Detectors per round: $n - k = 126$. Under an assumed 4 mechanisms per qubit per round (circuit-level noise; an assumption, not a published figure): 520 mechanisms/round, $\approx 1,040$ D-edges and ≈ 520 L-edges per round. At $R = 8$: nodes $\approx 126 \times 8 + 520 \times 8 + 4 = 5,172$; edges $\approx 1,560 \times 8 = 12,480$ — $\approx 2.9 \times$ the nodes ($5,172/1,779 = 2.907$) and $\approx 4.7 \times$ the edges ($12,480/2,681 = 4.655$) of the $d = 8$ surface-code DEM. The mechanism multiplier could plausibly range 2–8, giving a node-count range $\approx 2,600$ – $10,300$.

4.6 Isomorphism-invariance bound

A code-agnostic decoder is a function $f(G)$ on attributed DEM graphs. If two DEMs are isomorphic as attributed graphs, $f(G_1) = f(G_2)$ for any architecture using only graph structure and attributes — a representational ceiling. Conversely, distinct code families occupy distinct structural classes, forcing a single decoder to interpolate across classes. A cheap rigorous certificate: the part-size vectors (441, 1337, 1) for the $d = 8$ surface-code DEM and (126R, 520R, 4) for any R-round [[130,4,6]] DEM cannot coincide for integer R ($126R = 441$ gives $R = 3.5$; $520R = 1337$ gives $R = 2.57$; no integer satisfies both), so the two DEMs are provably non-isomorphic by part sizes alone. This is consistent with the reported cross-family generalization gap [1,2].

4.7 Cross-code model accounting (C4)

Input [1,2]: the cross-code model stays within 10% of BP+OSD on seen families. **Conditional:** this cross-code claim is not verifiable in the available abstract record of [1,2]. "Within 10%" means $pL^{\text{cross}} / pL^{\text{BP+OSD}} \leq 1.10$. Combined with C2's dedicated-model factor 0.84, the worst-case penalty of cross-code versus dedicated model is $1.10/0.84 \approx 1.31$ — a **conditional projection**, since the abstract does not state cross-code performance on [[130,4,6]] specifically. Honest bracket: [1.0, 1.31], conditional on the full preprint's cross-code evaluation tables.

5. Results

1. **Surface-code improvement factor:** 0.75 relative to correlated MWPM. Equivalent threshold improvement: $\approx 10.06\%$ at $d = 5$, $\approx 7.46\%$ at $d = 7$, $\approx 5.92\%$ at $d = 9$ (§4.1). Distance-equivalent at $p/p_{th} = 0.5$: exponent gain 0.4150374992788438, roughly half of one odd-distance step.
2. **qLDPC improvement factor:** 0.84 on $[[130,4,6]]$; equivalent threshold factor 1.0598398329483265 (illustrative). Encoding rate $0.03076923076923077 \approx 3.08\%$.
3. **Latency (conditional):** speedup bounded in $[4\times, 25\times]$, midpoint $\approx 10\times$; ≈ 286 shots/s/GPU vs ≈ 29 shots/s/core at midpoints; hardware-asymmetry caveat applies; conditional on latency claims not verifiable in the available abstract record (§4.3).
4. **Post-selection (conditional):** $>10\times$ error reduction at $>90\%$ retention would require discarded-shot failure rate $\approx 9.1\times$ base (§4.4); throughput cost $\leq 11.1\%$; accuracy-per-throughput gain $\geq 11.1\times$; conditional on claims not verifiable in the available abstract record.
5. **DEM sizing:** surface code $d = 8$, $R = 8$: 441 detectors, 1,337 mechanisms, 1,779 nodes, 2,681 edges, $O(d^2R)$ scaling (derived); $[[130,4,6]]$ at $R = 8$: $\approx 5,172$ nodes, $\approx 12,480$ edges (projection, stated assumptions).
6. **Non-isomorphism certificate:** surface-code and qLDPC DEMs provably non-isomorphic by part-size vectors for all integer R (§4.6).
7. **Cross-code penalty bracket:** $[1.0, 1.31]$ relative to the dedicated model (conditional projection on claims not verifiable in the available abstract record, §4.7).

6. Discussion

Limitations. (i) The threshold translations rest on the $\lfloor d/2 \rfloor$ ansatz, well supported for surface codes but heuristic for qLDPC; §4.2's figure is illustrative. (ii) All inputs come from the abstract of [1,2]; headline numbers are best-case ("up to 25%"), so typical-case performance may be materially lower. (iii) The latency bound compares GPU to CPU and therefore overstates any hardware-neutral advantage; a same-hardware decomposition is not derivable from published data. (iv) The qLDPC graph-size figures rest on an assumed mechanism multiplier; only the surface-code counts are fully derived, and even those depend on DEM construction conventions (first-round detectors, measurement-error indexing), which shift absolute counts by $O(R)$ but not the $O(d^2R)$ scaling. (v) The isomorphism argument uses part sizes — a weak invariant; the ceiling applies only to genuinely permutation-invariant architectures, which must be verified, not assumed.

Failure modes of the DEM-only paradigm. (i) *Size generalization:* the reported failure on unseen larger codes [1,2] is exactly what the graph-transfer literature [4] and the transfer-shift literature [8] predict: message-passing receptive fields are local, and larger codes exhibit correlation structures (e.g., logical-operator chains of length d) exceeding the receptive field of networks trained on smaller graphs. Scale-adaptive architectures [6] or hierarchical attention [13] are candidate but untested remedies. (ii) *Representational ceiling:* DEM-isomorphic codes are indistinguishable; structure beyond the DEM (e.g., gate-level scheduling) is unusable. (iii) *Density:* like the dense decoders critiqued in [7], the DEM graph is processed in full regardless of realized error rate; at low physical error rates a defect-centric sparse approach could dominate in wall-clock terms. (iv) *Training cost opacity:* the abstract implies retraining per DEM; "code-agnostic" describes the architecture, not the training cost, and total-cost comparisons must include training time, which is not reported.

Falsification experiments. (i) If the discarded-shot error rate in post-selection were below $\approx 9.1 \cdot p_L$ at 90% retention, the $>10\times$ claim would be arithmetically contradicted (§4.4). (ii) If BP+OSD ported to the same GPU matched POLYMECHANON's per-shot cost, the real-time advantage reduces to an implementation detail. (iii) If a

single model trained on small codes generalized to unseen large codes without size-conditioning, the structural-class account of the generalization gap (§4.6) would be falsified. (iv) If the 25% and 16% figures hold only at a single operating point, the threshold-equivalent translations would overstate the improvement. (v) A cross-family evaluation (train on small surface codes, test on large qLDPC without retraining) showing no significant degradation would refute the size-generalization concern; a significant one would quantify it.

Open questions. Can spectral invariants [11] serve as curriculum signals ordering DEMs by structural distance? Does the queuing-network view [3] yield an analytic baseline a GNN must beat to justify its cost? Can multi-view aggregation [9] or pretrain-then-finetune recipes [4,8] close the cross-code gap?

7. Conclusion

We have performed a transparent, arithmetic-driven assessment of the POLYMECHANON code-agnostic GNN decoder, converting its reported percentages into improvement factors (0.75, 0.84), threshold-equivalent shifts (≈ 6 –10% relative, distance-dependent), latency bounds ($[4\times, 25\times]$, midpoint $\approx 10\times$), post-selection separability requirements (discarded-shot rate $\approx 9.1\times$ base), and DEM graph sizes ($O(d^2R)$, with concrete counts). The DEM-as-input contract is a genuine architectural contribution; its principal open risks — size generalization, DEM fidelity, hardware-fair latency comparison, and unreported training cost — are each paired with a falsifiable test. Resolving size-generalization, possibly via scale-adaptive or hierarchical architectures, is the key condition for DEM-grounded decoding to become a route to universal decoders.

References

[1] TITLE: arXiv Query: `searchquery=&idlist=2610.01683&start=0&max_results=1` [2] A Code-Agnostic Graph Neural Network Decoder from the Detection Error Model. arXiv:2610.01683v1. <https://arxiv.org/abs/2610.01683v1> [3] A Tutorial about Random Neural Networks in Supervised Learning. arXiv:1609.04846v1. <https://arxiv.org/abs/1609.04846v1> [4] Learning Universal Graph Neural Network Embeddings With Aid Of Transfer Learning. arXiv:1909.10086v3. <https://arxiv.org/abs/1909.10086v3> [5] Graph Neural Networks for Channel Decoding. arXiv:2207.14742v2. <https://arxiv.org/abs/2207.14742v2> [6] GraphFPN: Graph Feature Pyramid Network for Object Detection. arXiv:2108.00580v3. <https://arxiv.org/abs/2108.00580v3> [7] Sparse Mamba Decoder for Quantum Error Correction: Efficient Defect-Centric Processing of Surface Code Syndromes. arXiv:2605.17156v2. <https://arxiv.org/abs/2605.17156v2> [8] Predicting concentration levels of air pollutants by transfer learning and recurrent neural network. arXiv:2502.01654v1. <https://arxiv.org/abs/2502.01654v1> [9] Mul-GAD: a semi-supervised graph anomaly detection framework via aggregating multi-view information. arXiv:2212.05478v1. <https://arxiv.org/abs/2212.05478v1> [10] DOI 10.5281/zenodo.18199940. QNFO: Comprehensive Technical Framework for Network Isomorphism. [11] DOI 10.5281/zenodo.18042721. QNFO: Topological Quantization and Spectral Filtration. [12] DOI 10.5281/zenodo.21566035. QNFO: Auditing the BQNN: Does a Tunable Quantum Neural Network on Trapped-Ion and Superconducting Hardware Demonstrate a Route to Near-Term Quantum Advantage?. [13] DOI 10.5281/zenodo.19648274. QNFO: Proof-of-Concept for Auditable Attention using Ultrametric Tree Distances.

Appendix A. Divergence report

D1. Representative decoding latencies (DIVERGENT). Draft A fixed $t_{GPU} = 5\text{ ms}$ and $t_{CPU} = 30\text{ ms}$, yielding a $6.0\times$ speedup; Draft B fixed $t_{GPU} = 3\text{ ms}$ and $t_{CPU} = 30\text{ ms}$, yielding $10\times$; Draft C used the full stated ranges, yielding $[4\times, 25\times]$ with midpoint $10\times$. Underlying convention: how to instantiate the qualitative phrases "a few ms"

and "a few tens of ms." **Resolution:** main text adopts Draft C's range-bounded convention ($[4\times, 25\times]$, midpoint $\approx 10\times$), since it makes no point choice within the stated ambiguity; Draft A's $6\times$ and Draft B's $10\times$ are interior point estimates consistent with this range.

D2. Absolute baseline logical failure rates (DIVERGENT). Draft A assumed baselines of 1.0×10^{-3} (surface code) and 2.5×10^{-3} (qLDPC) to produce absolute post-decoder rates (7.5×10^{-4} , 2.1×10^{-3}); Draft B declined to assume baselines, working purely with relative factors, because [1,2] does not report absolute baselines. **Resolution:** main text adopts Draft B's factor-based convention; absolute rates are omitted because the baselines are not source-traceable. Draft A's absolute figures remain valid conditional on its stated baseline assumptions but are not reproduced as headline results.

D3. Surface-code distance point for threshold translation (DIVERGENT). Draft A used $d = 5$; Draft B's primary computation used $d = 7$. **Resolution:** main text reports the translation at $d = 5, 7$, and 9 ($\approx 10.06\%$, $\approx 7.46\%$, $\approx 5.92\%$), documenting that the choice of distance point materially changes the threshold-equivalent figure; no single point is privileged.

D4. Post-selection modeling parameters (DIVERGENT). Draft A modeled a $12\times$ reduction with 92% retention (yielding effective error $\approx 6.8\times 10^{-5}$ under its assumed baseline); Draft B used the conservative $10\times$ and 90% reading and derived the discarded-shot rate $9.1\cdot p_L$; Draft C derived the throughput cost $\leq 11.1\%$ and figure of merit $\geq 11.1\times$. **Resolution:** main text adopts the conservative $10\times/90\%$ convention (Draft B) and integrates Draft C's throughput economics; Draft A's $12\times/92\%$ instantiation is a permitted but non-conservative reading and is not used.

D5. Scope of structural analysis (SINGLE, integrated). Draft C's DEM graph-size derivations and isomorphism certificate appear in no other draft; they are included in the main text (§4.5–4.6) as single-source contributions with their assumptions labeled. Draft B's cross-code penalty bracket (§4.7) is likewise single-source and included.

Appendix B. Claim attribution

Claim	Description	Source drafts	Status
C1	Up to 25% fewer logical failures than correlated MWPM on rotated surface code; factor 0.75	A, B, C	CONVERGENT
C2	Up to 16% fewer logical failures than BP+OSD on $[[130,4,6]]$; factor 0.84; encoding rate $\approx 3.08\%$	A, B, C	CONVERGENT
C3	Decoding time independent of physical error rate; few ms GPU vs tens of ms CPU	A, B, C	CONVERGENT
C4	Cross-code model within 10% of BP+OSD on seen families; fails on unseen larger codes	B, C	CONVERGENT
C5	Post-selection: $>10\times$ error reduction at $>90\%$ shot retention	A, B, C	CONVERGENT
C6	Speedup quantification: $6.0\times$ (A) vs $10\times$ (B) vs $[4\times, 25\times]$ (C)	A, B, C	DIVERGENT (D1)
C7	Absolute baseline logical failure rates assumed (1.0×10^{-3} , 2.5×10^{-3}) and resulting absolute rates	A	SINGLE (rejected in main text, D2)
C8	Threshold-equivalent translation via $[d/2]$ ansatz; distance-equivalent reading	B	SINGLE
C9	Post-selection separability requirement: discarded-shot rate $\approx 9.1\times$ base	B	SINGLE
C10	Post-selection throughput cost $\leq 11.1\%$; figure of merit $\geq 11.1\times$	C	SINGLE
C11	DEM graph-size derivations for surface code (441/1,337/2,681 at $d=8$, $R=8$) and qLDPC (projection)	C	SINGLE
C12	Isomorphism-invariance ceiling and part-size non-isomorphism certificate	C	SINGLE

Claim	Description	Source drafts	Status
C13	Cross-code penalty bracket [1.0, 1.31]	B	SINGLE
C14	Failure modes: generalization, DEM fidelity, training-data scarcity, hardware-asymmetric latency	A, B, C	CONVERGENT
C15	Falsification experiments (cross-family evaluation, DEM perturbation, same-hardware benchmark, scaling study)	A, B, C	CONVERGENT