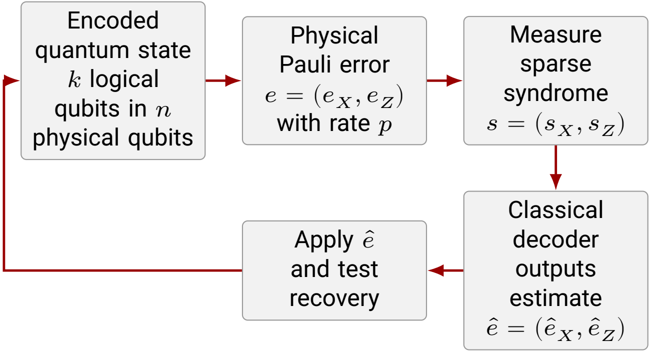
Reinforcement-Learning-Guided Multi-Branch Decoding of Quantum LDPC Codes

Vahid Nourozi[†,*] Toshiaki Koike-Akino[*] David G. M. Mitchell[†]

[†]New Mexico State University [*]Mitsubishi Electric Research Laboratories (MERL)

Basic QEC setup and notation



Define symbols before decoding

- e: actual physical error pattern.
- $\hat{e}$: correction estimated by the decoder.
- $e + \hat{e}$: residual error after applying the estimated correction.

A decoding attempt is successful if the residual $e + \hat{e}$ is **logically trivial**.

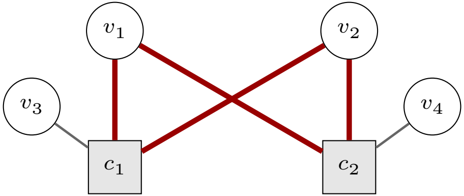
Syndrome-valid $\neq$ logically correct

For a CSS QLDPC code,

$$H_X H_Z^{\mathsf{T}} = 0, \qquad s_X = H_Z e_X^{\mathsf{T}}, \quad s_Z = H_X e_Z^{\mathsf{T}}.$$

CSS notation and syndrome equations follow Sec. II-A of the paper [1].

Why belief propagation struggles on QLDPC codes



A short cycle reuses correlated messages

Sparse Tanner graphs make BP attractive, but the tree-like independence assumption is weak at finite length.

Three quantum decoding obstacles

1. **Short cycles:** messages become correlated and can oscillate.
2. **Trapping configurations:** a small local pattern can prevent convergence.
3. **Degeneracy:** many physical corrections may represent the same logical action.

Scheduling is a control variable.

Which variable node is updated next can change the BP trajectory.

Poulin and Chung, Quantum Inf. Comput. 2008; Raveendran and Vasić, Quantum 2021; Moradi et al., 2026.

Key idea: do not trust one learned trajectory

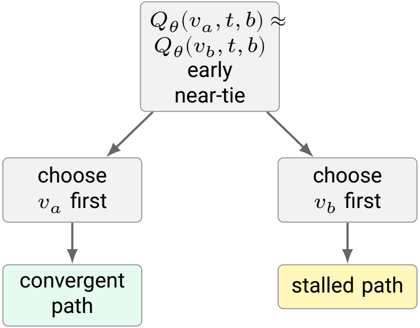
Learned scheduling score

A shared RL policy ranks eligible variable nodes using

$$Q_\theta(v,t,b) = \theta^\mathsf{T}\phi(v,t,b).$$

- v: candidate variable node (physical qubit) to update next.
- t: outer sweep index.
- b: branch index.
- $\phi(v,t,b)$: 28-dimensional feature vector.
- θ: learned 28-dimensional parameter vector.

The largest $Q_\theta(v,t,b)$ gets updated first.

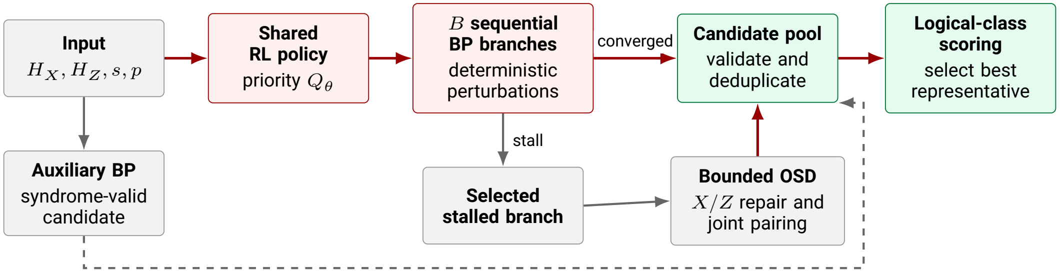


(schematic)

RL-MB-OSD

A learned trajectory plus reproducible nearby schedules.

RL-MB-OSD at a glance



Learned scheduling explores multiple trajectories; bounded OSD repairs selected stalls; logical-class scoring makes the final quantum decision.

A compact shared policy ranks every variable node

28 coefficients

shared across all variable nodes of one code matrix

$$Q_\theta(v,t,b) = \theta^\mathsf{T}\phi(v,t,b)$$

- No absolute node index; bounded features are clipped to $[0,1]$.
- One separately trained θ per code matrix; fixed during inference.
- Offline semi-gradient Q-learning rewards residual reduction and logical success.

Feature groups in $\phi(v,t,b)$

Features	Information available to the scheduler
2–8	Local unsatisfied-check fractions and H_X/H_Z interactions
9–12	Global residual-syndrome weights, mean, and imbalance
13–15	Sweep fraction, normalized channel rate, and quantum rate k/n
16–18	Current hard X/Z decisions and their product
19–21	Normalized X/Z reliability and joint unreliability
22–23	Recent hard-decision change and flip history
24–27	Variable degree and neighboring-check-weight summaries
28	Normalized branch index $b/(B-1)$

Exact entries are given in Table I of the paper [1].

Shared linear Q-learning follows [11]; branch perturbation and deterministic scheduling are from the present paper [1].

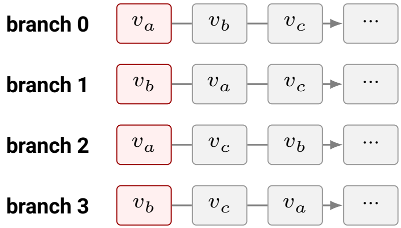
Deterministic branches explore nearby schedules

Perturbed priority

$$\widetilde{Q}_\theta(v,t,b) = Q_\theta(v,t,b) + \eta \frac{b}{B-1}\,\xi(s,b,t,v),$$

where $\xi \in [-1,1]$ is a deterministic hash of the syndrome, branch, sweep, and variable.

- Branch 0 follows the learned order exactly.
- Later branches increasingly reorder near-tied actions.
- Same input syndrome $\Rightarrow$ same trajectories.
- Each branch runs for at most T outer sweeps with early syndrome stopping.



reported setting: $B = 4, \eta = 0.35$

schematic node orders; actual orders are state dependent

T is a **per-branch sweep budget**; it is not directly comparable with every baseline's iteration count.

Bounded OSD repairs selected stalled branches

Componentwise syndrome repair

For a stalled hard decision $\hat{e} = (\hat{e}_X, \hat{e}_Z)$,

$$H_Z d_X^\mathsf{T} = s_X + H_Z \hat{e}_X^\mathsf{T}, \quad c_X = \hat{e}_X + d_X,$$

$$H_X d_Z^\mathsf{T} = s_Z + H_X \hat{e}_Z^\mathsf{T}, \quad c_Z = \hat{e}_Z + d_Z.$$

- d_X, d_Z: OSD repair vectors.
- c_X, c_Z: repaired component candidates.
- K_c: maximum retained candidates per component.
- K_p: maximum retained joint Pauli pairs.

Repair X component
$c_X^{(1)}, c_X^{(2)}, \dots, c_X^{(K_c)}$

Repair Z component
$c_Z^{(1)}, c_Z^{(2)}, \dots, c_Z^{(K_c)}$

Form Cartesian pairs
$(c_X^{(i)}, c_Z^{(j)})$
score by joint cost C_p

Keep at most K_p
syndrome-valid pairs

$o = 10 \Rightarrow$ **at most** $2^{10} = 1024$ **masks/component**

Reported limits: one OSD-eligible stalled branch, $K_c = 12$, $K_p = 64$.

C_p is the joint Pauli cost; $Y = (1, 1)$ is handled by the pair.

Bounded component-wise OSD and pair scoring are defined in Algorithm 2 and Sec. II-D of the paper [1]; OSD background: [12].

Quantum-aware selection scores logical classes

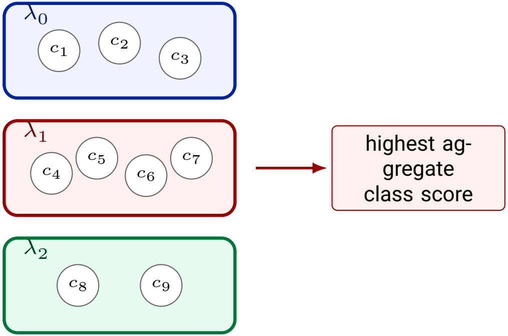
From candidates to a logical decision

1. Remove syndrome-invalid candidates.
2. Deduplicate exact corrections and retain multiplicity m_j.
3. Group candidates using their logical label λ_j.
4. Aggregate evidence within each class:

$$S(\lambda) = \mathrm{LSE}_{j:\lambda_j=\lambda}\left[-\frac{C_p(c_j)}{\tau} + \beta \log m_j\right].$$

Return the minimum-cost representative of $\arg\max_\lambda S(\lambda)$.

Reported values $\tau = 1$ and $\beta = 0.35$. Setting $\beta = 0$ removes multiplicity weighting.



Degeneracy becomes useful evidence: several low-cost representatives can support the same logical class.

Experimental setup and reproducibility

Reported configuration

Item	Setting
Codes	BB $[[144, 12, 12]]$ and A5 $[[180, 10, 15 \le d \le 18]]$
Channel	Code-capacity depolarizing noise
Training	$50{,}000$ episodes; $p \in \{.03, \dots, .07\}$; $T_{\mathrm{tr}} = 100$; seed 12
Sequential BP	$B = 4$; $\eta = 0.35$; normalized min-sum 0.625; LLR clip 25; early stopping
Repair and selection	$o = 10$; $K_c = 12$; $K_p = 64$; $\tau = 1$; $\beta = 0.35$
Auxiliary candidate	Syndrome-validated BP candidate with $T_{\mathrm{BP}} = 32$
Monte Carlo	Stop at 100 frame errors or 10^9 frames; report F/N and Wilson 95% interval

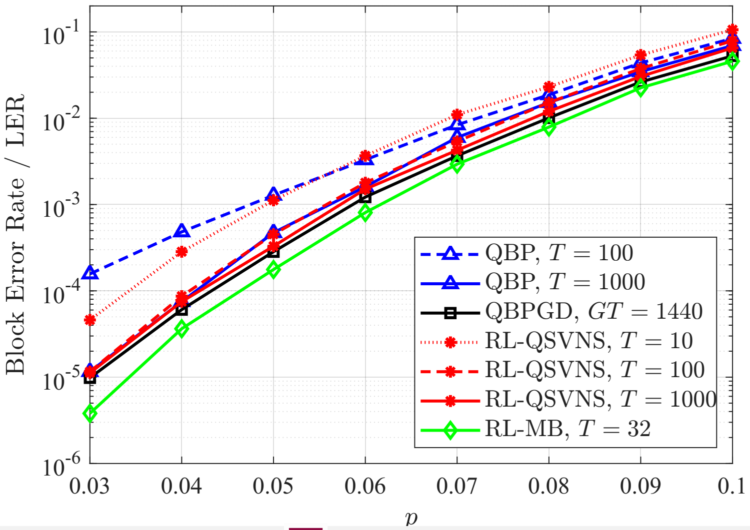
2 codes

BB and A5 policies are trained independently with the same 28-feature definition.

Comparison caveat
QBP, QBPGD, and RL-QSVNS are re-plotted from prior work [7], [9], [10]. They use neither common error frames nor matched computational budgets.

Interpretation: point-estimate comparison, not a paired or equal-latency test.

Result 1: $[[144, 12, 12]]$ bivariate-bicycle code



3.8×10^{-6}

RL-MB-OSD FER at $p = 0.03$
best re-plotted baseline: 9.8×10^{-6}

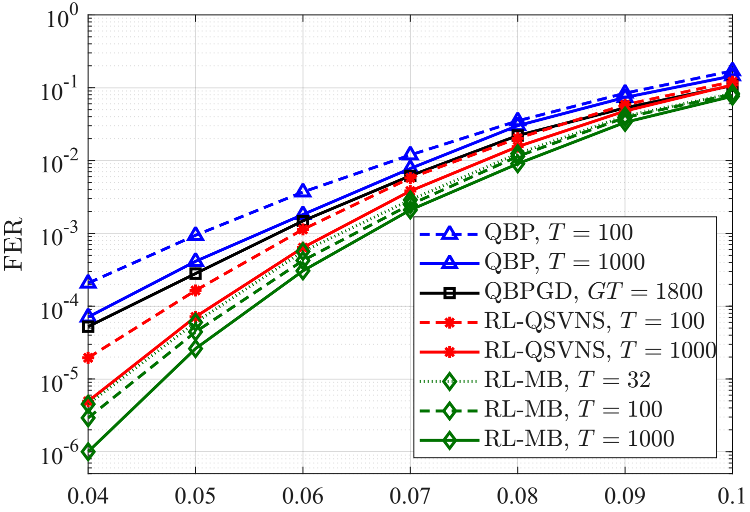
about $2.6\times$ **lower**

1.8×10^{-4}

RL-MB-OSD FER at $p = 0.05$
best re-plotted baseline: 2.8×10^{-4}

about $1.6\times$ **lower**

Result 2: $[[180, 10, 15 \leq d \leq 18]]$ A5 code



At $p = 0.04$

$T = 100 : 2.92 \times 10^{-6}$

95% CI $[2.40, 3.55] \times 10^{-6}$

$T = 1000 : 1.25 \times 10^{-6}$

95% CI $[1.02, 1.52] \times 10^{-6}$

$2.3\times$ lower

within-decoder FER estimate when the sweep budget increases from 100 to 1000

nonconvergences or invalid converged syndromes occurred at $p = 0.04$.

Complexity and remaining validation

Worst-case sequential work

Let $F = 28$ and $|E| = \mathrm{nnz}(H_X) + \mathrm{nnz}(H_Z)$. For B branches and T sweeps,

$$O(BT[|E| + Fn + n \log n]) \,.$$

Early stopping can reduce realized work.

Triggered OSD work

A repair call uses reliability sorting, elimination, and at most

$$N_{\mathrm{TEP}} = 2^{\min(o, q_{\mathrm{np}})}$$

test-error patterns per component before list limits.

Current curves do not yet prove

- equal-compute or equal-latency superiority;
- robustness across policy-training seeds;
- isolated gain from branching, OSD, or class aggregation.

Next decisive experiments

- matched-frame and matched-compute tests;
- ablations: $B = 1$, $\eta = 0$, no OSD, $\beta = 0$;
- mean/p95 latency and update counts.

Complexity and limitations follow Sec. IV of the paper [1].

Takeaways

1

Explore

A compact learned policy guides sequential BP, while deterministic branches explore nearby schedules reproducibly.

2

Repair and aggregate

Bounded OSD repairs selected stalls, and logical-class scoring uses quantum degeneracy rather than ignoring it.

3

Promising results

Lower FER point estimates are observed on BB and A5 under the reported settings; matched-compute validation is the next step.

Thank you — Questions?

Main claim: lower independently generated FER point estimates than the re-plotted baselines, not an equal-cost paired comparison.

References

1. V. Nourozi, T. Koike-Akino, and D. G. M. Mitchell, "Reinforcement-learning-guided multi-branch decoding of quantum LDPC codes," submitted manuscript, 2026.
2. N. P. Breuckmann and J. N. Eberhardt, "Quantum low-density parity-check codes," *PRX Quantum*, vol. 2, p. 040101, 2021.
3. D. Poulin and Y. Chung, "On the iterative decoding of sparse quantum codes," *Quantum Inf. Comput.*, vol. 8, pp. 987–1000, 2008.
4. N. Raveendran and B. Vasić, "Trapping sets of quantum LDPC codes," *Quantum*, vol. 5, p. 562, 2021.
5. J. Roffe, D. R. White, S. Burton, and E. Campbell, "Decoding across the quantum low-density parity-check code landscape," *Phys. Rev. Research*, vol. 2, p. 043423, 2020.
6. T. Hillmann *et al.*, "Localized statistics decoding for quantum low-density parity-check codes," *Nature Communications*, vol. 16, p. 8214, 2025.
7. H. Yao *et al.*, "Belief propagation decoding of quantum LDPC codes with guided decimation," in *Proc. IEEE ISIT*, 2024, pp. 2478–2483.
8. M. Ye, D. Wecker, and N. Delfosse, "Beam search decoder for quantum LDPC codes," arXiv:2512.07057, 2025.
9. M. Moradi, V. Nourozi, S. Habib, and D. G. M. Mitchell, "Learning to decode quantum LDPC codes via belief propagation," arXiv:2603.10192, 2026.
10. M. Moradi, S. Habib, V. Nourozi, and D. G. M. Mitchell, "Sequential BP-based decoding of QLDPC codes," arXiv:2602.13420, 2026.
11. C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, pp. 279–292, 1992.
12. M. P. C. Fossorier and S. Lin, "Soft-decision decoding of linear block codes based on ordered statistics," *IEEE Trans. Inf. Theory*, vol. 41, pp. 1379–1396, 1995.
13. P. Panteleev and G. Kalachev, "Degenerate quantum LDPC codes with good finite length performance," *Quantum*, vol. 5, p. 585, 2021.