

Review Article

Backlog Metastability in Windowed Quantum Error Correction Decoding

Jean-Jacques Dubray*

Cognitive Fab LLC, USA

*Corresponding author

Jean-Jacques Dubray, Cognitive Fab LLC, USA, Tel: 4255329561

Submitted: 21 August 2026

Accepted: 05 October 2026

Published: 06 October 2026

Copyright

© 2026 Dubray JJ

OPEN ACCESS

Keywords

• Surface Code; Real-Time Decoding; Minimum-Weight Perfect Matching; Fault-Tolerant Quantum Computing; State-Dependent Queues; Stochastic Stability; Correlated Error Bursts; Discrete-Event Simulation

Abstract

Real-time quantum error correction requires a classical decoder to consume syndrome measurements at the rate the quantum hardware produces them, roughly one round per microsecond on superconducting devices. The standard stability requirement is a utilization condition: the mean decode time per round must be less than the round period, $\rho < 1$. We show that this condition is inadequate for *windowed* decoders whose per-window cost grows superlinearly in the number of detection events, which is the empirical shape of matching-based decoders under burst-like syndrome content. For this class, stability is not a threshold but a *basin*. The deterministic backlog dynamics have a stable operating point and an unstable boundary B^* . Noise erodes that boundary downward, and sufficiently superlinear decoders escape spontaneously. In discrete-event simulations paired under common random numbers, a decoder with cost exponent $\gamma = 1.5$ and mean utilization $\rho_0 = 0.6$ (forty percent nominal headroom) fails without provocation, with a mean time to failure of $1,344 \pm 289$ rounds, about 1.3 ms of wall clock at a 1 μ s cycle. We measure the cost exponent of a production decoder (PyMatching 2, sparse blossom) and find it *anisotropic*: near-linear ($\gamma \approx 1.1$ – 1.2) when a window grows in length at fixed syndrome density, but approximately quadratic ($\gamma \approx 2$) when the same window grows denser. Bursts travel along the dangerous axis, where the basin is 2–4 rounds deep for ρ_0 between 0.5 and 0.8. Spatially *clustered* event content, the syndrome footprint of leakage and cosmic-ray bursts, is worse still: at matched event count, clustered windows cost $7.6\times$ length-grown ones and $4.8\times$ uniformly densified ones, so the quadratic density exponent is a conservative proxy for burst content. Greedy catch-up (“decode everything queued”) is precisely the divergent policy. A bounded decoding window restores stability, and the measured stability edge lands on a closed-form line. The simulation model is cross-validated against Stim at the level of individual fault mechanisms.

INTRODUCTION

A fault-tolerant quantum computation is a race between two processors. The quantum hardware emits syndrome measurements on a fixed cadence, and a classical decoder must convert them into corrections fast enough for the computation to proceed. If decoding falls behind, corrections arrive too late to steer non-Clifford operations, the logical clock stalls, and the backlog compounds. This compounding is the *backlog problem* [1-3]. The condition usually stated for real-time viability is a utilization bound: the mean decode time per syndrome round must be below the round period, $\rho = E[t_{\text{decode}}]/T_{\text{round}} < 1$. Real-time decoding demonstrations state their stability case in exactly these terms, certifying against the mean per-round latency [3,4].

Our claim is that for an important class of decoding systems the utilization bound is the wrong condition. It is not merely conservative or approximate. It is structurally wrong, in the sense that it cannot distinguish systems that differ by orders of magnitude in reliability. The class is that of *windowed* decoders with superlinearly event-dependent

cost: the decoder periodically takes a batch (window) of accumulated syndrome and pays a decode cost that grows faster than linearly in the number of detection events the window contains. Matching decoders have this shape in practice, and we measure it directly in Section 5.4.

Our contributions:

1. A basin, not a threshold (Sec. 2, 5.1). For window-cost exponent $\gamma > 1$, the deterministic backlog dynamics have two fixed points. The upper one, B^* , is an unstable basin boundary that no mean-based analysis can see. At a single mean utilization ($\rho_0 = 0.6$) we exhibit four decoders spanning unconditional stability to spontaneous failure.

2. Noise erodes the boundary (Sec. 5.1). The stochastic system diverges at perturbations the deterministic map survives. The deterministic reference is *drift-matched*: its batch costs use the same Poisson moments as the stochastic system’s expectations, so the gap is fluctuation physics rather than a Jensen mean-inflation artifact. The analytic B^* is therefore an *unsafe* provisioning line.

3. Spontaneous failure at nominal headroom (Sec. 5.2). At $\gamma = 1.5$ and $\rho_o = 0.6$, escape from the basin is driven by ordinary Poisson fluctuation alone. The measured MTTF is $1,344 \pm 289$ rounds, falling to 271 ± 33 at $\rho_o = 0.7$. The escape-time coefficient of variation is below one, so the failure is not memoryless, and MTTF alone under-describes it.

4. Greedy catch-up is the failure mode, and bounded windows are the fix (Sec. 5.3). With a per-window overhead, the per-round cost $c(W)$ is minimized at small windows, and the measured stability cliff lands on the closed-form line $c(W) = T_{\text{round}}$. The intuitive recovery policy, taking everything queued, is the divergent one.

5. The exponent of a real decoder is anisotropic (Sec. 5.4). Sparse blossom is near-linear when windows grow in *length* and approximately quadratic when they grow in *density*. At matched event count, density-packed windows cost $1.6\times$ length-grown ones, and spatially clustered content, the blob-shaped syndrome footprint that leakage and cosmic-ray events actually produce [5,6], costs $7.6\times$. Bursts deliver density, and clustered density is the worst case we measured.

Methodologically, all simulations run on a verification-first discrete-event engine (Sec. 4). Its runs are reproducible from content-addressed identities, its state machines carry runtime-checked invariants, and its paired comparisons are audited at the level of random draws. The underlying circuit model is cross-validated against Stim [7] mechanism for mechanism. The cross-validation caught three defects that internal verification could not (Sec. 4.2).

MODEL: WINDOWED DECODING AS A STATE-DEPENDENT QUEUE

Syndrome rounds arrive deterministically, one per period T . All times are in ticks, with $T = 10$ throughout, read as $0.1 \mu s$ per tick against a $1 \mu s$ cycle. Round r carries $w_r \sim \text{Poisson}(\lambda)$ detection events. We use $\lambda = 1.6$ as the model's reference rate; its physical anchoring is order-of-magnitude. Our cross-validated distance-3 memory at $p = 10^{-2}$ (Sec. 4.2) yields 1.59 detection events per three-round shot, which is 0.53 per round, while the distance-13 benchmark circuits of Section 5.4 carry 2.6 ($p = 10^{-3}$) to 5.2 ($p = 2 \times 10^{-3}$) events per round on their much larger detector set ($\lambda \propto d^2$ at fixed p). Measured hardware brackets the same range: the detection probabilities reported for the Willow surface codes imply roughly 0.6 to 4.2 events per round at $d = 3$ to 7 [3]. The value $\lambda = 1.6$ sits inside that physical range. Because c_l is calibrated to a target utilization at whatever λ is chosen (see below), λ sets the granularity of fluctuations rather than the mean

load, and burst sizes quoted in events should be read against the chosen λ . A 200-event burst, for example, is 125 round-equivalents at $\lambda = 1.6$. The decoder serves the queue in *windows*: when free, it removes up to W queued rounds (greedy: $W = \infty$) as one batch and occupies itself for

$$t(\text{batch}) = \max(1, \lceil c_0 + c_1 E^\gamma \rceil) \quad \text{ticks}, \quad (1)$$

where E is the total number of detection events in the batch. The one-tick floor matches the simulated machine, and the integer boundary matters for the $\gamma = 2$ entries of Table 1. Equation (1) is the declared cost model whose exponent Section 5.4 measures on a production decoder. We calibrate c_1 so that the steady-state single-round cost matches a target mean utilization: $c_0 + c_1 E[w^\gamma] = \rho_o T$.

The deterministic map. Under greedy windowing, a backlog of B rounds is served as one batch carrying $E \sim \text{Poisson}(B\lambda)$ events, and t/T new rounds arrive during the service. The deterministic (mean-drift) reference evolves the backlog by the *drift-matched* map

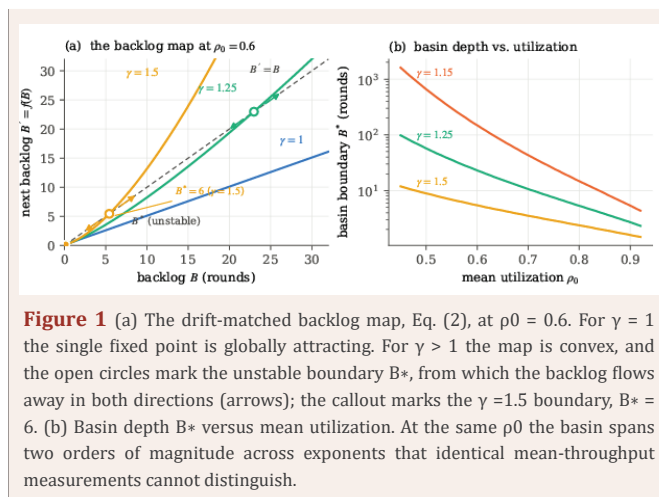
$$B' = f(B) = \frac{c_0 + c_1 \mathbb{E}[E^\gamma]}{T}, \quad E \sim \text{Poisson}(B\lambda). \quad (2)$$

Using the Poisson moment rather than the naive $(B\lambda)^\gamma$ keeps the reference system at the same mean drift as the stochastic one. With $w \equiv \lambda$ the reference would run at a lower effective utilization by Jensen's inequality, and part of any stochastic-versus-deterministic gap would then be a calibration mismatch rather than fluctuation physics. The correction is a few percent in the boundary region: the integer fixed points of Table 1 move by at most one round relative to the naive map, but the deterministic critical bursts of Figure 1 move by up to 20%.

For $\gamma = 1$, f is affine with a single, globally attracting fixed point whenever $\rho_o < 1$. This affine case is the textbook condition, and the queueing-literate reader will recognize it as valid precisely for *state-independent service*, which is the assumption that superlinear event-dependent cost violates. For $\gamma > 1$, f is convex and has *two* fixed points: a stable operating point B_o and an unstable boundary B^* . Any excursion beyond B^* , whether a burst or an unlucky run of fluctuations, diverges permanently, in a system whose mean utilization may be far below one. Table 1 shows how shallow the basin becomes as γ grows.

Table 1: Deterministic basin boundary B^* (first integer backlog outside the basin) from the drift-matched map (2), $\lambda = 1.6$, $T = 10$, $c_0 = 1$; " ∞ " = globally stable.

	$\rho_o=0.5$	0.6	0.7	0.75	0.8	0.9
$\gamma = 1$	∞	∞	∞	∞	∞	∞
$\gamma = 1.15$	658	148	44	25	15	6
$\gamma = 1.25$	58	23	11	8	6	3
$\gamma = 1.5$	9	6	4	3	3	2
$\gamma = 2$	4	3	2	2	2	2



RELATED WORK

Windowed decoding is as old as the surface code itself. Overlapping-window decoding appears already in Dennis et al. [8], and the modern real-time literature develops it into sliding-window and parallel-window architectures: Skoric et al.'s parallel window decoding [9], Tan et al.'s scalable window scheme [10], and Bombín et al.'s modular decoding [11]. Alongside these sit hardware decoders engineered for bounded latency, including LILLIPUT [12], Helios [13], and Barber et al.'s real-time collision-clustering decoder on FPGA [14]. The most complete demonstration to date is the real-time decoder of the Willow below-threshold experiment: streaming sparse blossom over ten-cycle processing blocks, with threads assigned to spacetime regions and fused asynchronously, holding a mean latency of $63 \mu\text{s}$ over 10^6 cycles at $d = 5$ against a $1.1 \mu\text{s}$ cycle [3]. Caune et al., integrate a sub-microsecond FPGA decoder directly into the control loop and report the backlog problem avoided, certified by mean per-round latency [4]. On the qLDPC side, Relay-BP targets real-time belief-propagation decoding of bivariate-bicycle codes [15], now demonstrated on FPGAs [16]; an iteration-bounded BP decoder caps its cost at an accuracy price, the same exchange Section 5.3 quantifies for windows. All of these architectures bound their windows by construction, historically motivated by latency and locality. Adaptive-window designs grow the window when decoder confidence drops [17]; expansion triggered by dense content is state-dependent service cost by construction, and without a bound equivalent to $c(W) < T$ it reinstates the convex coupling of Section 2 at the moment expansion is most expensive. System-level analyses of the real-time stack identify round time and tail latency, rather than mean throughput, as the binding constraints [18]. Our single-server greedy queue is the base case whose pathology explains, in dynamical-systems terms, why bounding the

window is not merely convenient but stabilizing: greedy catch-up is the divergent policy (Sec. 5.3), and a bounded window with $c(W) < T$ removes the unstable fixed point entirely. The backlog problem itself is stated in Terhal's review [1], and quantified by Smith et al. [2], at the utilization level. Our contribution is the structure below the utilization level.

The classical control side of the field is moving in a direction that makes this structure more binding, not less. HRL Laboratories has demonstrated a digitally controlled silicon quantum processing unit whose control electronics, including the syndrome-extraction sequencing for a distance-5 repetition code, run on a custom cryogenic CMOS controller inside the cryostat, with no real-time input from room-temperature equipment [19]. Decoding in that demonstration was a simple parity check evaluated offline. Section 6 returns to what the basin picture implies for decoders that must live within a cryogenic power budget.

On the queueing side, bistability and metastable congestion collapse in systems whose service degrades with state are classical subjects. Slotted ALOHA exhibits bistability [20], and loss networks exhibit metastability [21]; both show the two-fixed-point structure of Eq. (2), and that community will recognize the convex one-dimensional map on sight. The new content here is the QEC instantiation: the identification of the decoder's event-dependent cost as the state coupling, the measured anisotropy of that cost in a production decoder, and the resulting provisioning requirement. The excursion spectrum that drives escapes has direct physical referents. Cosmic-ray events produce catastrophic, spatially clustered error bursts in superconducting arrays [5], and leakage accumulates into correlated syndrome content [6]. On gap-engineered devices the measured spectrum has shifted but not vanished: the Willow repetition-code experiments observe correlated bursts about once per hour, localized to roughly 30 qubits, with detection probabilities elevated three- to six-fold and decaying over about $400 \mu\text{s}$ [3]. The residue has a measured mechanism: ionizing radiation transiently raises the quasiparticle density and produces correlated phase-error bursts lasting about a millisecond, which survive gap engineering [22]. All of these deliver exactly the dense, clustered windows that Section 5.4 measures as the most expensive per event. Correlated noise enters the decoding literature as an accuracy problem, because decoders built on independent-error models mismatch the true error distribution. The measurements of Section 5.4 give it a second consequence: clustered syndrome is also the most expensive content per event, so one physical footprint degrades accuracy and stability together.

METHODS

Verification-first simulation

All experiments run on polysim, a discrete-event simulation engine built around four disciplines that this study exercises end to end. First, every run is reproducible from a content-addressed identity (world hash, scenario hash, seed, replication index) and reproduces its journal digest byte for byte. Second, the simulated components are state machines with declared invariants checked on every transition; the decoder queue, for instance, carries machine-checked rules that its published backlog equals its actual queue and that a dropped round is a journaled refusal, never an omission. Third, paired comparisons run under common random numbers with a mechanical audit that the paired scenarios drew identical streams (every comparison below passed). Fourth, statistics that cannot be computed are refusals with named causes, never silently absent numbers. Across all runs reported in this paper the invariant monitors evaluated 21 declared rules per replication with zero violations.

Perturbations (bursts) are *scheduled*, not drawn, so the background noise realization is byte-identical across every scenario of a sweep. The exponent and calibration sweeps consume no random draws at all.

Cross-validation of the circuit model against Stim

The event rate λ and the burst semantics are grounded in a concrete code: a distance-3 rotated surface-code memory (9 data + 8 ancilla qubits, 3 rounds, ideal final readout) simulated by exact Pauli-frame propagation, with a generated layout, machine-checked schedules, and an exhaustive circuit-fault-distance proof used as a gate on all sampling experiments. The proof reproduces the known schedule-dependent fault-distance results, hook error witness included [23,24].

The model is cross-validated against Stim [7] at three levels. *Structural*: the single-fault symptom sets (which detectors fire, and whether the logical observable flips) of our mechanism table and Stim's detector error model are identical, with 199 distinct symptoms for the good schedule and 213 for the bad one, and with per-symptom probabilities agreeing to 1.9%, the $O(p)$ cross terms between channel composition and linear addition. *Distance*: Stim's undetectable-logical-error search returns minimum weights 3 and 2, matching our proofs. *Statistical*: 10^6 Stim shots decoded by our own lookup decoder give logical error rates 0.0283 / 0.0649 and detection fractions 0.0662 / 0.0657, each inside our engine's own intervals (0.0269 ± 0.0022 , 0.0656 ± 0.0034 , $0.0668/0.0665 \pm 0.0026$).

The cross-validation caught three defects that internal verification (invariants, known-answer tests, reproducibility) could not have caught, and we consider this the methodological finding of the exercise: a physically non-deterministic detector definition invisible to a Pauli-frame simulator by construction, a silent gap in a fault table labeled "proved," and a probe denominator off by exactly 24/20 (Appendix A details all three). Internal verification bounds self-consistency; only an external referee bounds truth.

RESULTS

Stability is a basin

Table 2 is the central result. Every cell has the same mean utilization ($\rho_0 = 0.6$), the same arrival process, and the same scheduled burst profile (40 rounds, beginning at round 400 of 4,000). Only the cost exponent and the burst magnitude vary. Divergence is classified as more than 100 rounds of outstanding work at the horizon. Recovered runs end near zero, diverged runs end in the hundreds to thousands, and the indicator is itself a probe whose ensemble mean carries a confidence interval.

First, mean-based provisioning cannot distinguish these four systems. At identical ρ_0 their behavior spans unconditional stability ($\gamma = 1$ absorbs a burst of $500\times$ the per-round load and recovers fully) to failure without

Table 2: P (divergence) after a scheduled burst of the given excess detection events ($n = 40$ replications per cell, $\rho_0 = 0.6$). "D" marks cells where the drift-matched deterministic map (2) also diverges. Percentages carry the Wilson 95% intervals drawn in Fig. 2; a 0/40 cell has a 95% upper bound of 8.8%.

burst $\rightarrow$	0	25	50	100	200	400	800
$\gamma=1$ ($B^*=\infty$)	0%	0%	0%	0%	0%	0%	0%
$\gamma=1.15$ ($B^*=148$)	0%	0%	0%	0%	8%	100% D	100% D
$\gamma=1.25$ ($B^*=23$)	0%	0%	43%	100% D	100% D	100% D	100% D
$\gamma=1.5$ ($B^*=6$)	100%	100%	100% D	100% D	100% D	100% D	100% D

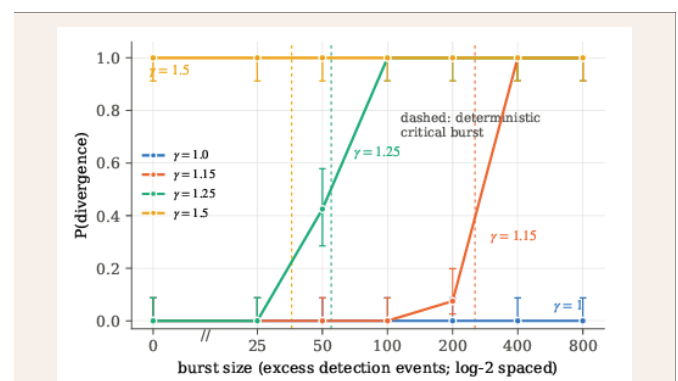


Figure 2 Measured P (divergence) versus burst size at $\rho_0 = 0.6$ ($n = 40$ per point; Wilson 95% intervals), with each exponent's *deterministic* critical burst as a dashed vertical line (bisection on the drift-matched noise-free dynamics). Every stochastic transition sits at or left of its deterministic line. Fluctuations carry the system over the boundary at perturbations the mean-field analysis survives, and $\gamma = 1.5$ requires no perturbation at all.

provocation. Second, *the stochastic boundary sits below the deterministic one*, and because the deterministic reference is drift-matched, the gap is genuinely fluctuation-activated rather than a mean-calibration artifact. The drift-matched critical bursts (bisection on the dynamics of Eq. (2); dashed lines in Figure 2) are 254, 55, and 36 events for $\gamma = 1.15$, 1.25, and 1.5. Against these, $\gamma = 1.25$ diverges in 43% of runs at a burst of 50, below its deterministic critical burst of 55, and $\gamma = 1.15$ diverges in 8% of runs at 200, which is 0.8× its deterministic critical burst. Provisioning against the analytic B^* therefore still fails stochastically. Third, the sharpest cell is $\gamma = 1.5$ at zero burst: 100% divergence in a system whose deterministic map survives bursts up to 36 events. This failure is noise-activated escape from a metastable basin, and we quantify it next.

Spontaneous escape is fluctuation-driven and sub-exponential

The failure rate is acutely sensitive to utilization, with a five- to forty-fold MTTF collapse per +0.1 of ρ_o . The measured $0.6 \rightarrow 0.7$ step is $1344/271 \approx 5\times$; the censored $0.5 \rightarrow 0.6$ step is at least $9\times$ and, under any reasonable extrapolation of the censored row, plausibly around $40\times$. The basin depth itself barely moves across the same range (Table 1), so the sensitivity lies in the escape rate, not the barrier position, as in thermal escape over a fixed barrier [7]. The total escape time's coefficient of variation is consistently below one: escape is *not* memoryless. A provisioning methodology that quotes MTTF alone therefore under-describes the failure process, because the distribution's shape matters.

Decomposing each escape path at the analytic boundary shows where the determinism lives. For each replication we record the first crossing of B^* and the time from that crossing to the 100-round threshold. The waiting time to first crossing is memoryless in both ensembles (CV = 1.05 at $\rho_o = 0.6$, 0.96 at 0.7; $n = 60$), consistent with noise-activated escape. The breakout leg behaves differently at the two utilizations. At $\rho_o = 0.7$ it is markedly narrower than exponential (219 ± 122 (sd) rounds, CV = 0.56), which is the wait-then-ramp picture. At $\rho_o = 0.6$, where B^* is only 6 rounds, the first crossing is a weak committor: the backlog brushes the shallow boundary and re-enters many times before a decisive excursion, so the first-crossing decomposition stays memoryless (CV = 1.09) even though the *total* escape time is sub-exponential (CV = 0.83). For basins this shallow, no single backlog level cleanly separates waiting from breakout. The sub-exponential total is the signature that does not depend on that choice.

Greedy catch-up is the failure mode

With a realistic per-window overhead ($c_o = 5$ of the

10-tick budget), the per-round cost of a width- W window $c(W) = c_o + c_l E[E]$ with $E \sim \text{Poisson}(W\lambda)$, is minimized at small W . Overhead dominates tiny windows, and the superlinear term dominates large ones. This section operates at $c_o = 5$ and $c_l = 1.17$, a drift-matched single-round steady cost of 7.9 ticks ($\rho \approx 0.79$).

This is deliberately hotter than the $\rho_o = 0.6$ used elsewhere, in order to place the window sweep across the stability line. For bounded W the map is asymptotically affine in B with slope below one whenever $c(W) < T$, so *bounding the window removes the upper fixed point entirely*. There is no basin boundary to fall over, and stability is global. Greedy windowing is what restores the convexity and, with it, the basin.

The condition $c(W) < T$ is necessary for stability but is not a performance guarantee. As $c(W) \rightarrow T$ the drain rate $T - c(W)$ vanishes and the recovery time after a burst diverges even on the stable side. Table 4 shows both regimes. Windows of 2–8 rounds absorb a 200-event burst with single-digit mean backlogs. The window $W = 16$ sits almost exactly on the line ($c(W) = 9.9$ against $T = 10$): it is stable in the strict sense, with no upper fixed point, but its post-burst drain is about 0.1 tick per round, and the “90%” entry is the more-than-100-outstanding-at-horizon classifier firing on a slow transient. A re-run at a 20,000-round horizon ($n = 30$) confirms the distinction. The $W = 16$ backlog peaks near 353 rounds and is draining by the horizon (ensemble mean 209, final 176), with 60% of replications still above the 100-round line; this behavior is *slow recovery*. The window $W = 32$, with $c(W) = 13.7 > T$, grows without bound (final backlog near 5,400 and rising); *this growth is instability*. The two regimes call for different

Table 3: Spontaneous escape at $\gamma = 1.5$, no perturbation ($n=60$, horizon 12,000 rounds; threshold 100 outstanding rounds). MTTF $\pm$ values here and throughout are 2 SE ($\approx 95\%$ CIs). The $\rho_o = 0.5$ row is right-censored: 20% escaped within the horizon, so the data bound the *median* escape time, not the mean.

	P (escape)	MTTF (rounds)	$\approx$ wall clock	CV
$\rho_o = 0.5$	20%	median > 12,000	> 12 ms	n/a
$\rho_o = 0.6$	100%	$1,344 \pm 289$	≈ 1.3 ms	0.83
$\rho_o = 0.7$	100%	271 ± 33	≈ 0.27 ms	0.47

Table 4: Bounded-window mitigation at $\gamma = 1.5$ under a 200-event burst ($n = 50$, horizon 6,000 rounds). The column $c(W)$ is the closed-form drift-matched per-round cost; the budget is 10. The classifier is >100 rounds outstanding at the horizon; for $W = 16$ this flags slow recovery, not instability (see text).

window	P (> 100 at horizon)	mean backlog (rounds)	busy	$c(W)$
$W = 1$	0%	4.0	87%	7.9
$W = 2$	0%	3.3	85%	6.3
$W = 4$	0%	5.3	86%	6.3
$W = 8$	0%	13.3	87%	7.5
$W = 16$	90%	215	98%	9.9
$W = 32$	100%	863	98%	13.7
greedy	100%	2,190	98%	(basin)

remedies: more margin in the first case, a different window in the second. The greedy policy, taking everything queued, which is the intuitive recovery move, diverges always, with a mean backlog three orders of magnitude above the bounded optimum.

The quantity $c(W)$ is a throughput criterion, and the optimum it selects ignores accuracy. Short windows harm matching accuracy at window boundaries, which is why practical designs use overlapping or fused windows of order the code distance, and why real-time operation carries a measured accuracy cost ($\Lambda = 2.0$ in real time versus 2.18 offline in the Willow experiment [3]). It therefore selects among accuracy-admissible windows; it does not license windows below the accuracy floor.

The measured exponent of a production decoder is anisotropic

The cost model (1) was declared; this section measures it. PyMatching 2 (sparse blossom [23]) decodes Stim-generated rotated-surface-code windows ($d = 13$). Decode time for individual shots, stratified by detection-event count and timed as a minimum over repeated calls, is regressed against events, with γ fitted by grid search (closed-form least squares in (c_0, c_1) at each candidate γ , SSE on *linear* time) and reported with a 5%-SSE band. Because a three-parameter power law trades c_0 against γ at the low- E end, we re-fit with c_0 pinned to an $E \rightarrow 0$ extrapolation as a robustness check. The exponents move from 1.14 to 1.16 (window) and from 2.31 to 2.05 (density), so the anisotropy is not a fit degeneracy.

The exponent depends on *how* the events arrive:

- **Window axis** (the window grows in rounds at fixed density, which is ordinary catch-up): $\gamma \approx 1.1$ – 1.2 , bands [1.01, 1.35] across passes, $R^2 \approx 0.97$. Near-linear.
- **Density axis** (fixed 40-round window, syndrome packed denser by raising p): $\gamma \approx 2$, bands spanning [1.7, 2.5] across passes, $R^2 \approx 0.95$. Approximately quadratic.
- **Cluster axis** (fixed window, fixed background $p = 10^{-3}$, with the k detectors nearest a random space-time center flipped, the syndrome footprint of a localized burst [5]): a global power-law fit is poorly conditioned here, because decode time at fixed E varies severalfold with blob location (boundary versus bulk), so we report the matched-event form below.
- **Matched-event check** (robust to the thermal

drift that moves the absolute fits; all three axes measured adjacently in one process): at 600–900 events, density-packed windows cost **1.6×** length-grown ones ($1.52\times$ in an earlier pass), and *clustered* windows cost **7.6×** length-grown ones, which is $4.8\times$ the uniformly densified windows at the same event count.

The mechanism is consistent with the algorithm's structure. In blossom-type matching, growing the window in length adds graph volume while defect interactions stay local, whereas packing defects closer multiplies alternating-tree and blossom interactions per defect. That is a plausible route to near-quadratic scaling, and clustering then concentrates the interactions in one region. In Figure 4, the density series' larger intercept (about $80 \mu s$ versus about $20 \mu s$ at small E) reflects the fixed 40-round graph paying its size at low occupancy, while the window series builds small graphs at small E . At matched E the two sweeps also differ in graph size (a 160-round sparse window versus a 40-round dense one), so the $1.6\times$ ratio folds together density and graph-size effects. The cluster series, which holds both the graph and the background fixed, isolates content structure, and it is the cleanest evidence that burst-shaped syndrome is intrinsically expensive.

Mapping the measured exponents through Eq. (2) gives Table 5, where $\gamma = 1.09$ and 1.23 are the two independent passes' point estimates and 1.35 is the upper edge of the pooled 5%-SSE band. Steady-state catch-up sits in a deep basin, thousands of rounds at $\gamma = 1.09$. But within the window-axis confidence band alone, B^* at $\rho_0 = 0.6$ ranges from effectively infinite to 11 rounds: *real-time provisioning requires pinning the cost exponent to about ± 0.1* , a requirement we have not seen stated anywhere in the literature. Burst content, on the density axis, faces a basin of 2–4 rounds for ρ_0 between 0.5 and 0.8. A sparse-blossom-class decoder provisioned on its steady-state throughput curve therefore meets bursts on a cost curve two exponent classes worse, and the cluster measurement says even that is the optimistic reading.

On this machine, single-core per-round decode time corresponds to $\rho \approx 2.0$ ($d=13, p=10^{-3}$) up to $\rho \approx 4.8$ ($d=21$) against a $1 \mu s$ cycle. One core cannot hold real time at all, which is why real systems window and parallelize, and

Table 5: Basin boundary B^* (rounds) at the *measured* exponents, drift-matched map.

	$\rho_0=0.5$	0.6	0.7	0.8
window axis, $\gamma = 1.09$ (pass 1)	38,331	3,211	423	75
window axis, $\gamma = 1.23$ (pass 2)	79	30	13	7
window axis, $\gamma = 1.35$ (band edge)	20	11	6	4
density axis, $\gamma \approx 2$	4	3	2	2

why the windowing *policy* is the operative lever. This does not contradict the published sparse-blossom throughput of roughly $1\ \mu\text{s}$ per round at $d = 17$ and $p = 10^{-3}$ [23]. That figure is batched C++ throughput, while ours is *per-call* invocation through the Python boundary, whose constant lands in c_0 and whose per-shot dispatch forgoes batching. Per-call is the operationally relevant mode for real-time windowing: a window must be decoded when its deadline arrives, not when a batch fills.

DISCUSSION

For windowed decoders with superlinearly event-dependent cost, *report the basin, not the utilization*. A provisioning claim should state (i) the measured cost exponent on both axes with a confidence band; (ii) the implied B^* at the operating utilization; (iii) the expected excursion spectrum, that is, burst sizes and rates from the device's leakage and radiation environment, for which measured magnitudes exist [3-22]; and (iv) the window bound W placing $c(W)$ safely below the round period, with margin, since the recovery time diverges as $c(W) \rightarrow T$ even on the stable side. Items (i) and (iv) are a measurement and a design choice, item (ii) is closed-form, and item (iii) is the one input this paper takes from outside. The newest decoder implementations already report worst-case rather than mean latency [25], the reporting posture that items (i) through (iv) systematize. Parallel-window architectures [9-11] satisfy (iv) by construction; this paper supplies the dynamical-systems reason that the construction is load-bearing. Code class moves the exponent as well: noise-biased cat-qubit architectures suppress bit-flips passively and reduce active correction to a repetition code [26], whose quasi-one-dimensional matching graph should sit near $\gamma = 1$ on all three axes, placing that architecture, pending measurement at its operating density, in the unconditionally stable row of Table 1. The basin is a property of the code and decoder pair, not of quantum error correction as such.

The Willow below-threshold experiment is the existence proof that the mitigation regime is practical [3]. Its real-time decoder never executes greedy catch-up: syndrome is processed in bounded blocks and fused asynchronously, a working instance of the bounded-window rule, and it holds its mean latency constant over a million cycles at $d = 5$. The same data shows the tail our analysis predicts: infrequent latency spikes above 1 ms, roughly nine hundred cycles of outstanding work, absorbed and drained on the stable side. The spontaneous failures of Section 5.2 are a property of the greedy policy class, not a prediction about deployed systems that already bound their windows. The open question is margin, not

existence. Syndrome volume grows quadratically with distance and decoder latency grows with code size, as that experiment's own outlook notes, and the basin accounting above is the instrument for deciding how much margin the next factor of d^2 requires. A burst of the magnitude and duration measured above carries roughly 1,500 to 2,500 excess detection events, so real excursions do not probe the edge of any basin in Table 2; they overrun it outright. Surviving them is therefore entirely a question of the drain rate $T - c(W)$ against a once-per-hour recurrence, which is Section 5.3's regime.

HRL Laboratories has reported a digitally controlled silicon quantum processing unit: a 54-dot array configured as up to 18 exchange-only spin qubits, operated entirely by a custom cryogenic CMOS controller inside the cryostat, executing a distance-5 repetition code over hundreds of autonomous syndrome-extraction rounds with sub-microsecond operations [19]. That demonstration solves, at the chip level, exactly the half of the real-time problem this paper takes as given: syndrome generation and control at rate, with all control signals generated inside the cryostat. Readout in that system is still digitized at room temperature, and decoding was a simple parity check evaluated offline; real-time decoding at error-correction speed remains open on that platform. Wherever the future decoder lands, the constraint is the same. On the room-temperature side it inherits the latency of the link that carries the syndrome out of the refrigerator. Inside the cryostat it inherits a hard power and area budget (the reported controller draws 3.5 W at 4 K), and a cryogenic decoder cannot buy its way out of a backlog excursion with burst compute, because the compute ceiling is fixed at design time. In that regime the windowing policy is not one lever among several; it is the lever. The provisioning discipline above then becomes a design input, since nothing can be added after the cryostat closes. The demonstration also shows that sub-microsecond operation is no longer specific to superconducting hardware, so the round period in our model should be read as a platform parameter rather than a superconducting constant.

Two features cut against intuition. Greedy catch-up, the natural recovery behavior and the default of any decoder that simply processes what has accumulated, is exactly the policy that converts a transient overload into a permanent one. And the deterministic basin boundary, though closed-form and correct as far as it goes, is not a safe engineering line: fluctuations carry the system over it from below, at rates our escape-time measurements quantify.

What this model deliberately omits marks the next experiment. The feedback studied here is *computational*:

backlog leads to costlier windows. The *physical* feedback, in which a stalled logical clock idles qubits and raises the syndrome density the decoder then faces, compounds along the measured density axis, which is the quadratic one. The HRL experiment measures this ramp directly: without leakage-reduction units the detector event fraction grows steadily with syndrome round, and only their inclusion holds it stationary [19]. In our terms, leakage mitigation is what keeps λ constant; its absence hands the decoder a rising $\lambda(r)$ under which the operating point and the basin boundary drift toward each other and annihilate. We expect the coupled feedback to shrink the basin further. Modelling it requires connecting the queue back into the circuit-level noise model, which our simulation framework supports but this paper does not yet exercise.

LIMITATIONS

The cost curve (1) is a declared model. The calibration (Sec. 5.4) measures one decoder (PyMatching 2.4.0), on one machine (a laptop-class x86 core), for one code family, with the Python call boundary contributing a constant to c_0 . The density-axis exponent moves by roughly ± 0.3 with thermal state, and the matched-event ratio is the stable form of that claim. The cluster measurement flips *detectors* in a space-time ball, the syndrome footprint of a localized burst, rather than simulating a microscopic burst mechanism; it isolates what blob-shaped syndrome costs the matcher, not how a specific physical event produces it. The cost model applies to decoders whose batch grows with backlog: per-call windowed invocation, catch-up after stalls, and adaptive-window designs. Streaming architectures that hard-cap the decode problem on a constant-size graph buffer [3] have already applied the mitigation of Sec. 5.3; for them the state dependence moves into queueing delay rather than service cost, and they inherit the slow-recovery regime rather than the basin. The model's event rate $\lambda = 1.6$ is an order-of-magnitude anchor between the measured distance-3 rate (0.53/round) and the benchmarked distance-13 rates (2.6–5.2/round), and it is treated as backlog-independent. The arrival process is Poisson with independent rounds. Measured devices show correlated fluctuations beyond this, including unexplained spikes in detection-event rates on a silicon platform [19], and catastrophic correlated bursts on superconducting ones [5]; because escape rates are exponentially sensitive to fluctuation scale, an overdispersed arrival process at the same mean would shorten the MTTFs of Table 3, so those values should be read as optimistic for equal mean rate. Divergence is classified against a declared threshold (100 outstanding rounds) at a declared horizon. This classifier flags slow recovery as well as true instability, and Sec. 5.3 disambiguates the two by horizon-doubling

where it matters. The escape-time probe shares the same threshold by construction. The $\rho_0 = 0.5$ escape row is right-censored at the simulation horizon. The distance-3 circuit model, not the queue, is what the Stim cross-validation covers; the queue model's external referent is the decoder benchmark. Sampling grids use $n = 40$ –60 replications per cell. Sequential stopping was not applied to the basin grids, and the sharp 0/100 cells are limited by that n accordingly.

CONCLUSION

For windowed decoders whose cost grows superlinearly in syndrome content, real-time stability is a basin, not a threshold. The utilization bound $\rho < 1$ cannot separate systems that differ by orders of magnitude in reliability: at a single mean utilization we exhibit decoders spanning unconditional stability to spontaneous failure, show that fluctuations carry the system over a drift-matched boundary the mean-field analysis survives, and measure the failure clock at 1.3 ms of wall time with forty percent nominal headroom. The exponent that controls all of this is a measurable property of a real decoder, and it is anisotropic: bursts meet sparse blossom on a cost curve two exponent classes worse than steady-state catch-up, and clustered burst content is more expensive still. The design consequence is to bound the window, place $c(W)$ below the round period with margin, and report the basin alongside the utilization.

The framework is built to carry the next set of measurements. Arrival dispersion comes first: the MTTFs reported here assume Poisson arrivals, the most benign case, and escape rates are exponentially sensitive to fluctuation scale, so MTTF as a function of measured dispersion is the next number a provisioning methodology needs. Drifting load follows: unmitigated leakage hands the decoder a rising event rate under which the operating point and the boundary annihilate, and the extinction time is closed-form and testable against the ramps hardware already measures [19]. The exponent taxonomy should widen beyond matching: union-find and clustering decoders, and the quasi-one-dimensional repetition strips of noise-biased architectures [26], at their operating densities, plausibly occupy the stable row of Table 1 and should be measured there. Neural decoders belong in the same taxonomy and sit at its stable end by construction: a network with fixed input dimension and constant-time inference does bounded work per window, which is the $\gamma = 1$ row [27]. Hybrid designs qualify that placement, since a pre-decoder that routes hard instances to a matching fallback [25–29], inherits the matching exponent on precisely the dense windows that trigger the fallback, and its effective exponent follows the fallback path rather than

the fast one. Finite decode memory converts escape from a latency failure into a data-loss rate with its own placement rule. And the loop this paper left open, a stalled logical clock raising the syndrome density the decoder then faces, closes through the circuit-level noise model along the quadratic axis. The direction of the hardware gives these measurements their urgency: as classical control moves into the cryostat and compute ceilings become design constants, the basin stops being an operational tuning concern and becomes an input to the machine's blueprint.

Reproducibility and data availability

The results in this paper occupy three tiers of reproducibility. The deterministic results (Tables 1 and 5, the $c(W)$ column of Table 4, and the critical bursts of Figure 2) are closed-form consequences of Eq. (2) and the constants stated with each table, and can be recomputed from this paper alone. The decoder measurements (Sec. 5.4, Figure 4) depend only on the open tools Stim and PyMatching; the benchmark script and the measured points behind Figure 4 accompany this paper as a standalone artifact. The stochastic simulations (Tables 2-4, Figure 3) were produced on polysim, a proprietary discrete-event engine. Within it, every run is addressed by a content-addressed identity (world hash, scenario hash, seed 20260817, replication index) and reproduces its journal digest byte for byte; paired comparisons carry a machine audit that scenarios consumed identical random streams; probes and their thresholds are content-hashed into each experiment's identity. The per-replication outputs behind those tables and figures (divergence indicators, escape times, breakout decompositions) accompany this paper as data files, and the queue model is specified completely by Eqs. (1) and (2) together with the parameters listed in each section, so it can be reimplemented in any discrete-event simulator.

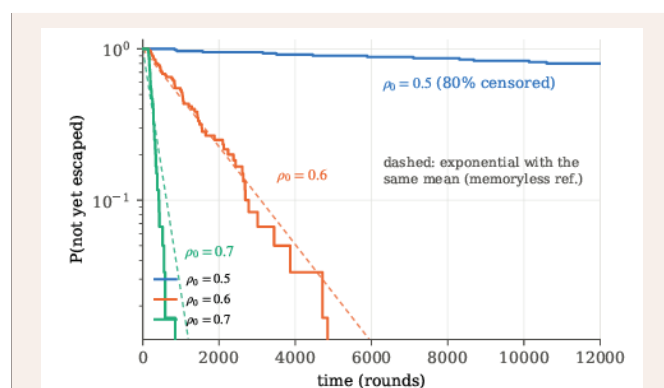


Figure 3 Escape-time survival curves at $\gamma = 1.5$, no perturbation ($n = 60$ each; the $\rho_0 = 0.5$ curve is right-censored at the horizon). Dashed: exponential references with the matching means. The measured curves are convex relative to memoryless decay, with few early escapes and then a faster-than-exponential fall. This shape is the $CV < 1$ signature of a waiting time followed by a near-deterministic breakout.

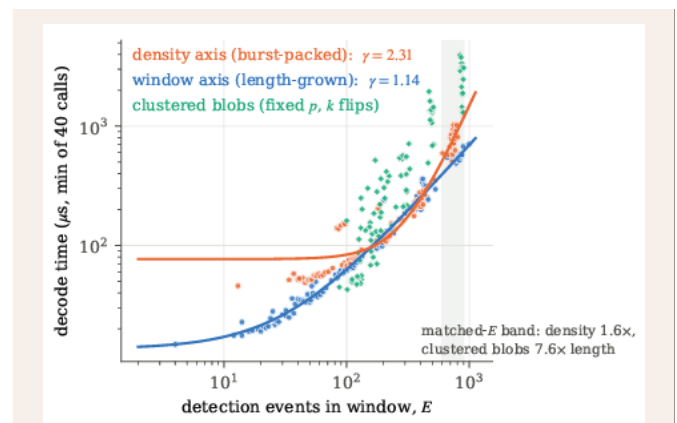


Figure 4 Measured decode time of PyMatching 2 versus detection events per window ($d = 13$; each point the minimum of repeated calls). Growing the window in length at fixed density (blue) fits $\gamma = 1.14$ in this pass; packing the same window denser (orange) fits $\gamma = 2.31$. Both are within the multi-pass bands quoted in the text. Green diamonds: the cluster axis, a fixed 40-round window at fixed background $p = 10^{-3}$ with k spatially clustered detector flips added, sitting well above both fits at the same event count. The shaded band marks the matched-event comparison: at identical E , density-packed windows cost $1.6\times$ and clustered windows $7.6\times$ length-grown ones.

A The three cross-validation defects

Each defect below survived the engine's internal verification (21 machine invariants evaluated per replication, known-answer tests, byte-exact journal reproducibility) and was caught only by the external cross-validation against Stim.

a) Physically non-deterministic detectors. A boundary-plaquette mirroring in the layout generator produced a collision-free, invariant-clean circuit whose detectors compare measurements that are not deterministic in the absence of errors. A Pauli-frame simulator is *constitutively* unable to see this failure class: it tracks deviations from a reference execution it never checks, so a detector that is broken in the reference looks exactly like a working detector with a relabeled baseline. Stim's structural comparison (which detectors fire for which fault) exposed it immediately.

b) A silent gap in a table labeled "proved." The fault enumeration used one schedule variant's gate slots when enumerating both variants, leaving mechanisms of the second schedule unenumerated. The gap was invisible to internal consistency checks because the enumeration is itself the reference the checks run against. The cross-validation's symptom-set equality (199 and 213 distinct symptoms) failed until the enumeration was made schedule-dependent.

c) A denominator off by exactly 24/20. A detection-fraction probe divided events over the machine's 20 detector components rather than the 24

syndrome bits of the readout convention. The defect was found because Stim's detection fraction refused to match ours by a suspiciously clean ratio. Ratios of small integers in a discrepancy are a fingerprint of denominator disagreement, not physics.

REFERENCES

1. Terhal BM. Quantum error correction for quantum memories. *Rev Mod Phys*. 2015; 87: 307-346.
2. Satvik Maurya, Swamit Tannu. Managing classical processing requirements for quantum error correction. *arXiv: 2406.17995*. 2024.
3. Google Quantum AI and Collaborators. Quantum error correction below the surface code threshold. *Nature*. 2025; 638: 920-926.
4. Caune L, Skoric L, Blunt NS, Ruban A, McDaniel J, Valery JA, et al. Demonstrating real-time and low-latency quantum error correction with superconducting qubits. *Nat Commun*. 2026; 17: 7383.
5. Matt McEwen, Lara Faoro, Kunal Arya, Andrew Dunsworth, Trent Huang, Seon Kim, et al., Resolving catastrophic error bursts from cosmic rays in large arrays of superconducting qubits. *Nat Phys*. 2022; 18: 107-111.
6. Kevin C. Miao, Matt McEwen, Juan Atalaya, Dvir Kafri, Leonid P. Pryadko, Andreas Bengtsson, et al. Overcoming leakage in quantum error correction. *Nat Phys*. 2023; 19: 1780.
7. Gidney C. Stim: a fast stabilizer circuit simulator. *Quantum arXiv: 2103.02202*. 2021; 5: 497.
8. Dennis E, Kitaev A, Landahl A, Preskill J. Topological quantum memory. *J Math Phys*. 2002; 43: 4452.
9. Skoric L, Browne DE, Barnes KM, Gillespie NI, Campbell ET. Parallel window decoding enables scalable fault tolerant quantum computation. *Nat Commun*. 2023; 14: 7040.
10. Tan X, Zhang F, Chao R, Shi Y, Chen J. Scalable surface-code decoders with parallelization in time. *PRX Quantum*. 2023; 4: 040344.
11. Bombín H, Dawson C, Liu YH, Nickerson N, Pastawski F, Roberts S. Modular decoding: parallelizable real-time decoding for quantum computers. *arXiv: 2303.04846*. 2023.
12. Das P, Locharla A, Jones C. LILLIPUT: a lightweight low-latency lookup-table decoder for near-term quantum error correction. *arXiv: 2108.06569*. 2022.
13. Liyanage N, Wu Y, Deters A, Zhong L. Scalable quantum error correction for surface codes using FPGA. *QCE*. 2023; 1: 916-927.
14. Barber B, Barnes KM, Bialas T, Bugdayci O, Campbell ET, Gillespie NI, et al. A real-time, scalable, fast and resource-efficient decoder for a quantum computer. *Nat Electron*. 2025; 8: 84-91.
15. Müller T, Alexander T, Beverland ME, Bühler M, Johnson BR, Maurer T, et al. Improved belief propagation is sufficient for real-time decoding of quantum memory. *arXiv: 2506.01779*. 2025.
16. Maurer T, Bühler M, Kröner M, Frank H, Tristan M, Drew V, et al. Real-time decoding of the gross code memory with FPGAs. *arXiv: 2510.21600*. 2025.
17. Oberoi T, Vizslai J, Chong FT. ADaPT: adaptive-window decoding for practical fault-tolerance. *arXiv: 2605.01149*. 2026.
18. Chen YJ, Luan CY, Zheng P, Zeng X, Hou J, Fu Z, et al. Real-time quantum error correction system stack: architecture, algorithms, and engineering practice. *arXiv: 2605.30765*. 2026.
19. Members of the HRL Quantum Team and Collaborators. A digitally controlled silicon quantum processing unit. *Nature*. 2026; 655: 1154.
20. Gibbens RJ, Hunt PJ, Kelly FP. Bistability in communication networks, in *Disorder in Physical Systems*. Oxford Univ. Press. 1990; 113-127.
21. Kleinrock L, Lam SS. Packet switching in a multiaccess broadcast channel: performance evaluation. *IEEE Trans Commun*. 1975; 23: 410.
22. Kurilovich VD, Roberts G, Martin LS, Matt M, Alec E, Laraet F, et al. Correlated phase error bursts in a gap-engineered superconducting qubit array. *Phys Rev*. 2026; 16: 021025.
23. Higgott O, Gidney C. Sparse Blossom: correcting a million errors per core second with minimum-weight matching. *arXiv: 2303.15933*. 2023.
24. Hirai Y, Ikari S, Ueno Y, Suzuki Y. No More Hooks in the Surface Code: Distance-Preserving Syndrome Extraction for Arbitrary Layouts at Minimum Depth. *Quant-ph*. 2026.
25. Wegmann P, Augoustis T, S'wierkowska A, Giortamis E, Bhatotia P. Zero-G: a pre-decoder-aware decoder for quantum error correction. *arXiv: 2608.02030*. 2026.
26. Lescanne R, Villiers M, Peronnin T, Sarlette A, Delbecq M, Huard B, et al. Exponential suppression of bit-flips in a qubit encoded in an oscillator. *Nat Phys*. 2020; 16: 509.
27. Bausch J, Senior AW, Heras FJH, Edlich T, Davies A, Newman M, et al. Learning high-accuracy error decoding for quantum processors. *Nature*. 2024; 635: 834-840.
28. Mullan M, Weippert M, Brown W. Improved methods for determining quantum error correcting code performance and fault tolerance. *arXiv: 2607.27153*. 2026.
29. Kramers HA. Brownian motion in a field of force and the diffusion model of chemical reactions. *Physica*. 1940; 7: 284-304.