RTM-Aware Quantum Computing

A Multiscale, Slope-First Framework for Coherence, Scheduling, and Design

Álvaro Quiceno

Abstract

We introduce a **slope-first** methodology for quantum computing based on **Multiscale Temporal Relativity (RTM)**. Inside a fixed operational regime, RTM posits that a characteristic time T scales with a size/scale proxy L by a power law,

$$\log T = \alpha \log L + c,$$

where the **coherence exponent** α is the **clock-invariant** structural signal and c encodes clock/units. We adapt RTM to quantum stacks—**physical**, **QEC**, **compiler/runtime**, and **I/O–cryo**—by defining layer-specific (L, T) pairs (e.g., number of active qubits vs. stable calibration time; code distance vs. logical-failure time; multiplexing degree vs. readout latency; circuit width vs. makespan), and estimating binwise slopes under errors-in-variables (ODR/TLS, Theil–Sen, SIMEX). A **collapse test** validates scaling and guards against regime mixing; clean family-wise slopes are fused into a real-time $\mathbf{ECI}_{QC}(\mathbf{t})$ with uncertainty and QA gates.

We formulate **falsifiable** hypotheses: **(H1)** higher pre-shock α predicts longer stability margins (fewer forced recalibrations, lower logical error at fixed d); **(H2) decoherence events**—significant QA-clean drops in ECI_{QC}—lead spikes in logical error, queueing, or makespan; **(H3)** micro→meso→macro **tempo cascades** exhibit non-decreasing α within stable regimes. We demonstrate how **RTM-aware scheduling** (batching, staggered resets, low-variance routing), **QEC cadence design** (desynchronization of syndrome cycles), and **modular sizing** (sweet spots for interconnect) can improve throughput and reliability without changing physical fidelities. The framework is reproducible, gauge-robust (unit/clock changes do not affect α), and designed to fail gracefully (no-collapse and high heterogeneity become scope boundaries, not post-hoc fixes).

Preliminary empirical validation⇒(APPENDIX G). We validate the RTM diagnostic framework in quantum hardware through a systematic analysis of 31 IBM Quantum processors spanning 5 to 1121 qubits. Initial raw scaling analysis suggested a positive coherence-to-size relationship ($\alpha = +0.23$); however, RTM isolates this as a statistical illusion driven by a manufacturing confounder (generational technology improvements). When controlling for technology generation, the true topological scaling reveals a strictly negative exponent of $\boldsymbol{\alpha \approx -0.35}$. This places macroscopic quantum decoherence unequivocally in the **Inverse Transport Class** ($\alpha < 0$), alongside classical Stokes-Einstein diffusion. This empirical result proves that as quantum system size (L) increases, topological noise (crosstalk, correlated defects) scales collectively rather than independently, causing the system to decohere faster. RTM successfully separates

underlying physical scaling laws from engineering artifacts, demonstrating that massive coherence requires architectural resonance, not merely brute-force monolithic scaling.

1. Introduction

1.1 Motivation: beyond fidelities and error rates

Quantum performance is usually summarized by **point metrics**—single- and two-qubit fidelities, T_1/T_2, logical error rates, or benchmark figures (QED-C, QV). Yet practical reliability and throughput hinge on something orthogonal: **how timing stretches across scale** in a multistage stack—qubits and resonators, code cycles, compilers, cryogenic I/O. When small subsystems respond quickly and larger ones respond more slowly in a disciplined, layered fashion, shocks are **dissipated**; when timings **flatten**, disturbances percolate across layers and synchronize failures (stalling readout, spiking logical error, or forcing global recalibrations).

Multiscale Temporal Relativity (RTM) provides a compact language for this phenomenon. Inside a fixed regime, RTM expects a power-law relation between a **characteristic time** T and a **scale proxy** L: the **slope** α in log $T = \alpha$ log $L + c$ is structural (invariant to time units), while the intercept c is a **clock** (gauge). We bring this principle to quantum computing and show that measuring, validating, and **engineering** α yields actionable levers—independent of nominal units—to improve stability and throughput.

1.2 RTM in one line

Structure lives in the slope; clocks live in the gauge.
A change of clock or units shifts c but leaves α unchanged. Thus α can be compared across devices, stacks, and labs, while c cannot.

1.3 Contributions

This paper makes five contributions:

1. **Operationalization of RTM for QC.** We define layer-specific (L, T) pairs for **physical**, **QEC**, **compiler/runtime**, and **I/O–cryo** layers (e.g., $L =$ active qubits, $T =$ stable calibration time; $L = d$, $T =$ cycles to logical failure; $L =$ multiplexing degree, $T =$ readout latency; $L =$ circuit width, $T =$ makespan).

2. **Validation & estimation.** We provide a **collapse test** (residual independence of log $T - \alpha$ log L from log L) to detect regime mixing and non-power curvature, and adopt **errors-in-variables** estimation (ODR/TLS, Theil–Sen, SIMEX) with bootstrap uncertainty and changepoint guards.

3. **A single real-time indicator.** We fuse family-wise slopes into $ECI_{QC}(t)$ via random-effects meta-analysis with heterogeneity controls $(Q, I^2, \hat{\tau}^2)$; we publish QA flags and withhold fusion when proxies disagree.

4. **Design levers.** We formalize **RTM-aware scheduling** (batching, staggered resets, low-variance routing), **QEC cadence design** (desynchronization to avoid phase lock between physical errors and syndrome extraction), and **modular sizing** (choosing module/interconnect scales that elevate α without throttling throughput).

5. **Falsifiable hypotheses & protocols.** We pre-register **H1–H3** with A/B protocols on superconducting and trapped-ion platforms, metrics (throughput, makespan, logical error, uptime, p95/p50 ratios), and decision thresholds for adoption.

1.4 What α is—and is not

- **Is:** a **binwise slope** linking a time T to a scale L inside a **fixed environment** (same temperature/firmware/topology/syndrome schedule). It captures the **geometry of tempo across scale**.

- **Is not:** a causal parameter by default; level changes in T (units, clocks, offsets) do **not** change α. When collapse fails, α is **undefined** for that bin and should not be fused.

1.5 Layer-specific (L, T) exemplars (preview)

- **Physical:** L = active qubits / coupler degree / cluster size; T = stable calibration interval, gate/RO latency, mean time to drift.

- **QEC:** $L = d$ (code distance) or number of logical qubits; T = cycles to logical failure; cadence of syndrome extraction.

- **Compiler/runtime:** L = circuit width or depth after mapping; T = makespan; queueing delay and rescheduling latency.

- **I/O–cryo:** L = multiplexing degree or channels; T = readout latency/BER recovery; p95 queue length.

Each layer yields a slope $\hat{\alpha}_f$; after QA and collapse, we fuse them into $ECI_{QC}(t)$ with uncertainty bands. Clean **decoherence events** are significant drops in ECI_{QC} over pre-registered horizons.

1.6 Hypotheses (falsifiable)

- **H1 (Resilience):** Higher pre-shock α associates with smaller logical-error spikes at fixed d and longer stable calibration intervals.

- **H2 (Anticipation):** QA-clean ECI_{QC} drops lead increases in makespan, queueing, or logical error by weeks to months, adding predictive value over baselines (fidelity, utilization, temperature).

- **H3 (Cascade):** Within stable regimes, $\alpha_{physical} \leq \alpha_{QEC} \leq \alpha_{runtime/I/O}$; directionality tests favor micro→meso→macro timing flow.

1.7 RTM-aware design (intuitions we will test)

- **Scheduling:** Avoid patterns that **flatten** α (long, tightly coupled operations in parallel); favor **batching** readouts and **staggered** resets to prevent synchronization cascades.

- **QEC cadence:** Introduce slight **desynchronization** (phase offsets) between syndrome cycles and known noise rhythms to raise α_{QEC}.

- **Modularity:** Choose module size and interconnect density where α is high enough to damp inter-module cascades but not so high that throughput is throttled.

1.8 Relation to prior work

Our framework complements fidelity-centric and error-model approaches by adding a **scale–tempo geometry**. It is compatible with (not a replacement for) surface/LDPC code theory, compilation/routing heuristics, and queueing models; it contributes a **gauge-invariant** statistic α and a **collapse** specification test to separate **structure** from **clock** effects. In the language of stochastic processes, our dynamics section (later) connects RTM to **time-changed diffusions**; in meta-analysis terms, our fusion mimics **random-effects** with explicit **heterogeneity gates**.

1.9. Systematic Empirical Validation: The Illusion of Monolithic Scaling⇒(APPENDIX G)

A fundamental premise of RTM is its ability to diagnose the true transport class of a system by observing its scaling exponent α. In the race to build fault-tolerant quantum computers, hardware developers have continuously scaled up monolithic processor sizes (qubit counts). Superficially, historical data seems to suggest that larger processors possess better coherence times (T_2). However, within the RTM framework, we must ask: is this improvement a property of the spatial scale (L), or is it an artificial offset generated by continuous technological advancements?

To answer this, we utilize RTM as a diagnostic filter on 31 IBM Quantum processors. We hypothesize that quantum decoherence is not a set of isolated independent events, but a collective topological collapse. Therefore, true physical scaling should exhibit an **Inverse** transport signature (α < 0), where a larger geometric footprint naturally amplifies

crosstalk and correlated noise. By isolating technology generations, we demonstrate how RTM cuts through manufacturing confounders to reveal the stark, underlying physics of macroscopic quantum systems.

2. RTM Foundations Adapted to Quantum Computing

This section states the RTM axioms, derives the **power-law** form $T = \kappa L^\alpha$, and tailors **clock/gauge** and **collapse** notions to quantum stacks. Throughout, $L > 0$ is a **scale proxy** (layer-specific) and $T > 0$ is a **characteristic time** measured in a **fixed environment/bin** (same temperature, firmware, topology, syndrome schedule, utilization band).

2.1 Axioms (binwise)

A1 — Scale semigroup. For any dilation $b > 0$,

$$T(bL) = f(b)\, T(L),$$

with $f(1) = 1$ and $f(b_1 b_2) = f(b_1) f(b_2)$.

A2 — Mild regularity. f is measurable (or continuous at $b = 1$).

A3 — Clock invariance in-bin. Allowed **clock changes** multiply T by a factor $c > 0$ **independent of** L inside the bin (unit changes, timestamp baselines, fixed-latency offsets). In QC practice: rescaling time units, constant readout overheads, constant cryo I/O baselines.

A4 — Binning. Comparisons are made within bins where environment is stable. If a changepoint is detected, the bin must be split.

2.2 Functional-equation solution → power law

Let $u = \log\, L, v = \log\, T$. From A1–A2, the multiplicative Cauchy equation gives $f(b) = b^\alpha$ for some $\alpha \in \mathbb{R}$. Hence

$$T(L) = \kappa L^\alpha, v(u) = \alpha u + \log\,\kappa.$$

Interpretation. α is the **coherence exponent** (slope); κ is a **clock** (intercept).

2.3 Clocks (multiplicative gauge vs. additive latency)

In RTM, a "clock change" inside a fixed bin is a **multiplicative** rescaling of all characteristic times: $T' = cT, c > 0$ independent of L. This includes time-unit conversions (ns↔μs), uniform timebase/tick-rate rescalings, or uniform calibration factors. In log coordinates, $\log T' = \log T + \log c$, so α is unchanged and only the intercept shifts.

By contrast, **constant latencies** (e.g., fixed readout preamble, pipeline delay, timestamp baseline offsets) are **additive**: $T_{\text{obs}} = T + b$. On log–log plots this is not a pure intercept shift and can bias α, especially when T is not $\gg b$. Therefore, before estimating α, either:
(i) estimate/subtract the latency b and fit using $T_{\text{eff}} = \max(T_{\text{obs}} - b, \varepsilon)$, or
(ii) restrict analysis to regimes where $T_{\text{obs}} \gg b$ and report sensitivity of α to plausible b.

2.4 Collapse as a binwise specification test

Given observations $\{(L_i, T_i)\}_i$ in a bin, define $x_i = \log L_i, y_i = \log T_i$. Fit a binwise slope $\hat{\alpha}$ (Section 5) and examine **residuals**

$$\tilde{y}_i := y_i - \hat{\alpha} x_i.$$

Collapse test. In a valid RTM bin, $\tilde{y}$ should be **independent of** x (up to noise). We operationalize with:

- A regression $\tilde{y} \sim x$ and require $R^2_{\text{collapse}} < \tau$ (default $\tau = 0.05$).

- A **clock placebo**: multiply all T_i by a constant; $\hat{\alpha}$ and R^2_{collapse} must be unchanged.

- A **smooth check** (LOESS or spline) for visible trend; if present, reject the bin.

Meaning. Collapse establishes that, after removing $\hat{\alpha} \log L$, only a **gauge** remains (intercept noise), not a trend vs. scale.

2.5 Variable exponents and finite-window bias

In practice, α can drift slowly with environment or scale (e.g., across utilization bands or multiplexing factors). Write

$$v(u) = \int_{u_0}^{u} \alpha(s)\, ds + \log \kappa(u),$$

with $|\alpha'(u)| \leq \varepsilon$ small on the window and κ **slowly varying**. For any symmetric window of width h in u,

$$\hat{a}(u; h) \;=\; \alpha(u) \;+\; O(\varepsilon h) \;+\; O(\text{slow-variation}),$$

and

$$R^2_{\text{collapse}} \;=\; O((\varepsilon h)^2).$$

Rule. Choose bins/windows small enough that curvature is negligible; otherwise split the bin.

2.6 Failure modes (should fail)

RTM is designed to **predict its own failure**:

1. **Regime mixing (kinks).** Example: changing the readout chain or syndrome scheduler mid-bin. The log–log plot shows a slope change at $L^\star$; collapse fails.

2. **Curvature (non-power).** Example: a multiplexing-dependent overhead that grows nonlinearly with L. Residuals trend with x; collapse fails even after rebinning.

3. **Scale-dependent clocks.** Any "clock" factor $c(L)$ that depends on L is not a gauge; it injects du-components into the 1-form and must be modeled explicitly (or the bin rejected).

2.7 QC layer mapping (notation and exemplars)

We will use these **canonical** (L, T) pairs in later sections (others may be added if they pass collapse):

- **Physical**:
  $L =$ number of **active qubits** (or cluster/coupler degree);
  $T =$ **stable calibration interval**, **gate** latency, **readout** latency, or **mean time to drift**.

- **QEC**:
  $L =$ **code distance** d (or logical-qubit count);
  $T =$ **cycles to logical failure** at fixed target error.

- **Compiler/Runtime**:
  $L =$ **circuit width** or **post-mapping depth**;
  $T =$ **makespan** or **queueing delay**.

- **I/O–Cryo**:
  $L =$**multiplexing degree** or readout-channel count;
  $T =$**effective readout latency / BER-recovery half-life / p95 queue length (in time)**.

Each family produces a binwise $\hat{\alpha}_f$. Only families that **pass collapse** and QA contribute to the fused indicator $\boldsymbol{ECI}_{QC}\textbf{(t)}$ (Section 6).

2.8 Why α matters operationally

- **Comparability**: α is invariant to unit changes and constant overheads, enabling **cross-lab** and **cross-generation** comparison.

- **Early warning**: significant **drops** in α (per family or fused) signal **decoherence events** likely to precede spikes in logical error, makespan, or forced recalibrations.

- **Design lever**: raising α (without over-layering) via **scheduling**, **QEC cadence**, or **module sizing** improves damping of cross-scale cascades.

2.9 Summary

RTM in QC reduces to three binwise statements: (i) **power-law** scaling $T = \kappa L^\alpha$, (ii) **gauge invariance** (only the slope α is structural), and (iii) **collapse** as a falsifiable specification test. With careful binning and EIV-aware estimation, α becomes a reproducible, unit-robust **coherence exponent** that guides both **diagnostics** and **design** across the quantum stack.

---

3. Scale–Clock Geometry for QC (Collapse as Exactness)

We recast RTM for quantum stacks in geometric form. The key object is the **RTM 1-form**

$$\omega \;=\; d(\log T) \;-\; \alpha(x)\, d(\log L),$$

defined on a bin E with **environment** coordinates x (temperature, firmware, topology, syndrome schedule, utilization band) and **scale** $u = \log L$. In this language, **collapse** is equivalent to **exactness/flatness** of ω; regime seams and non-power curvature appear as **holonomy/curvature**. This section states the results and instantiates them with QC failure modes.

3.1 Spaces, bins, and the RTM 1-form

- **State space.** $M = X \times \mathbb{R}$ with coordinates (x, u), where $u = \log L$.

- **Clock potential.** $v(x, u) = \log T(x, L)$.

- **RTM 1-form.** $\omega = dv - \alpha(x)\, du$ (constant-α case) or $\omega = dv - \alpha(x, u)\, du$ (slow drift allowed).

A **clock change** (unit/baseline shift independent of L inside a bin) is $v \mapsto v^{\#} = v + \phi(x)$. Then

$$\omega \;\mapsto\; \omega^{\#} = \omega + d\phi(x)$$

— a **gauge transformation** by an exact 1-form pulled back from X. Hence α **is gauge-invariant**.

3.2 Collapse $\Leftrightarrow$ exactness/flatness

Theorem 3.1 (Collapse $\Leftrightarrow$ exactness).
On a simply connected bin E, the following are equivalent:

1. (RTM chart) $v(x, u) = \alpha(x)\, u + \log \kappa(x)$ (or $v = \int \alpha(x, s)\, ds + \log \kappa(x)$ for slow drift).

2. (**Collapse**) Residual $\tilde{v} := v - \alpha u$ is independent of u in E.

3. (**Exactness**) $\omega = d\psi$ on E for some $\psi(x)$ (no u-dependence).

Corollary 3.2 (Flatness test).
$d\omega = 0$ is necessary and (on simply connected E) sufficient for collapse. With $\alpha = \alpha(x, u)$,

$$d\omega \;=\; -\, d\alpha \wedge du.$$

Thus curvature (non-power behavior) or regime mixing gives $d\alpha / du \neq 0$ and **breaks collapse**.

3.3 Holonomy and regime seams (QC failure modes)

Define **holonomy** around a closed loop $\gamma \subset E$: $\mathcal{H}(\gamma) = \oint_{\gamma} \omega$. If $\mathcal{H}(\gamma) \neq 0$, collapse cannot hold globally.

QC instances.

- **Scheduler seam.** Changing the syndrome-extraction cadence mid-bin (new FPGA image) produces a kink in $v(u)$; loops that cross the seam pick up nonzero holonomy → **rebin**.

- **Readout chain swap.** A per-channel overhead that *depends on multiplexing* behaves like a scale-dependent clock $c(L)$; this is **not gauge** and injects du-components → collapse fails (and should).

- **Thermal drift window.** A slow utilization ramp changes α across u; if $\partial_u \alpha$ is not small on the window, $d\omega \neq 0 →$ split the bin or shrink the window.

3.4 Adiabatic collapse (slowly varying α)

If $|\partial_u \alpha| \leq \varepsilon$ on a window of width h,

$$\tilde{v}(x, u) = v - \alpha(u_0, x) u = \log \kappa(x) + O(\varepsilon h),$$

and the empirical collapse statistic obeys

$$R^2_{\text{collapse}} = O\left((\varepsilon h)^2\right).$$

Practice. Choose h so that $\varepsilon h \ll 1$; otherwise, reduce the bin or model the drift explicitly.

3.5 Morphisms (reparametrizations) and gauge

Let $\Phi = (\varphi, \psi)$ map $(X_A, L_A, v_A) \to (X_B, L_B, v_B)$, where $\varphi: X_A \to X_B$ reparametrizes environment and $\psi: X_B \to \mathbb{R}$ is a clock change. Then

$$\Phi^* \omega_B = \omega_A + d(\psi \circ \varphi).$$

Interpretation: transporting the structure from B to A preserves **slope** and alters only the **clock** by an exact form. This formalizes cross-lab/device comparisons when units/baselines differ.

3.6 Diagnostics and acceptance gates (QC checklist)

1. **Collapse test.** Fit $\hat{\alpha}$ (Section 5), compute residuals $\tilde{y} = y - \hat{\alpha} x$; require $R^2_{\text{collapse}} < 0.05$ **and** no trend in a nonparametric smooth.

2. **Clock placebo.** Multiply all T by a constant; $\hat{\alpha}$ and R^2_{collapse} must be unchanged.

3. **Changepoints.** Run detectors on (x, y) and on $\tilde{y}$; any kink $\Rightarrow$ rebin.

4. **Window control.** Ensure $|\partial_u \alpha|\ h$ is small (adiabatic regime).

5. **Publish/withhold.** Only bins passing 1–4 contribute to $\text{ECI}_{\text{QC}}(t)$; otherwise label `NO_COLLAPSE` or `REGIME_MIX`.

3.7 What this buys us operationally

- A **proof-obligation**: show flatness/exactness (collapse) before trusting a slope.

- A **debugger**: nonzero holonomy localizes seams (scheduler swaps, readout changes).

- A **tuning rule**: reduce h or rebin until $d\omega \approx 0$; if impossible, the domain is **non-power**—treat α as undefined there.

3.8 Summary

The scale–clock geometry makes two RTM statements precise for QC:

1. α **is a gauge-invariant structural quantity**, unaffected by unit/baseline changes;

2. **Collapse equals exactness/flatness of** ω, and its failure is informative (curvature or seams).
   We will now leverage this to define **operational** (L, T) (Sec. 4) and to estimate $\hat{\alpha}$ robustly under measurement error (Sec. 5).

4. Operational (L, T) Definitions and Binning Protocol

This section turns RTM into **measurable practice** for quantum stacks. We define layer-specific (L, T) pairs, specify **sampling**, **units**, and **guards**, and give a binning protocol that avoids regime mixing. Throughout, $u = \log\ L, v = \log\ T$.

4.1 Design principles for (L, T)

- **One mechanism per family.** Each (L, T) pair should reflect a single dominant mechanism (e.g., readout pipeline, not a mixture of readout + routing).

- **Monotone** L. L should increase with "problem size" at that layer (width, distance, channels, cluster size).

- **Clock independence.** Within a bin, **multiplicative** timebase changes ($T' = cT$) are allowed gauges (unit/timebase rescalings). **Additive** overheads ($T_{obs} = T + b$) must be subtracted, modeled, or avoided (fit only where $T \gg b$); otherwise they may bias slopes and invalidate collapse.

- **Steady sampling.** Use **fixed cadence** collection; record raw timestamps to allow reslicing.

4.2 Physical layer

Candidates for L:

- $L =$ number of **active qubits** in the workload window;

- $L =$ **cluster size** (connected qubits participating simultaneously);

- $L =$ **coupler degree** (average fanout).

Candidates for T:

- **Stable calibration interval** (time until any qubit in the cluster exits tolerance);

- **Gate latency** (median single/two-qubit gate duration across the active set);

- **Readout latency** (median per-shot time to valid symbol under fixed thresholds);

- **Mean time to drift** (MTTD) for frequency/phase.

Instrumentation.

- Log per-shot timestamps; a calibration watchdog recording when thresholds are breached; attach environment tags: temperature band, firmware hash, bias point.

Non-examples.

- Mixing *both* gate latency and readout latency in the same T.

- Letting L be "qubits defined on chip" (not necessarily active).

4.3 Error Correction (QEC)

L: code **distance** d (primary), or number of **logical qubits** at fixed d.
T:

- **Cycles to logical failure** at a fixed target error (median or survival quantile);

- **Syndrome-cycle latency** (mean time per cycle under fixed schedule).

Scheduling notes.

- Freeze a **syndrome schedule** (FPGA image + cadence). Any change $\Rightarrow$ new bin.

- Record bias (X/Z) and leakage mitigation settings.

Edge cases.

- If T is dominated by **rare catastrophic events** (e.g., resonator latch-ups), prefer **conditional medians** (exclude known catastrophic flags) and report a sensitivity panel.

4.4 Compiler / Runtime

L: circuit **width** (max concurrent qubits) or **post-mapping depth**; optionally **active layers** after routing.
T:

- **Makespan** (submission $\rightarrow$ completion);

- **Queueing delay** (submission $\rightarrow$ start);

- **Rescheduling latency** after a calibration event.

Controls.

- Fix **routing policy** and **placement heuristic** inside a bin.

- Stratify by utilization band (e.g., 0–30%, 30–60%, >60%). If utilization drifts, split the bin.

4.5 I/O – Cryo / Readout

L: **multiplexing degree** (channels per line) or number of concurrent readout channels.
T:

- **Readout latency** (median p50 and tail p95);

- **BER recovery half-life** after a controlled burst;

- **Queue p95** expressed in time.

Instrumentation.

- Timestamp every DMA/ADC burst; log per-channel buffers; annotate firmware versions of DSP.

Caveat.

- Per-channel overheads that **grow with** L are *not* gauges; they are genuine scale effects—permissible for RTM—but if the overhead itself changes mid-bin, collapse should fail and trigger a split.

4.6 Binning protocol (environment fixing)

A **bin** is a maximal interval where the environment is effectively constant.

Bin key (example):

BIN
= {platform, temperature band, firmware hash, topology ID, routing policy, syndrome cadence, utilization band}.

Procedure.

1. **Slice** data by BIN; discard slices with $<N_{\min}$ distinct L values (default 6).

2. **Changepoint scan** on $y = \log T$ vs. $x = \log L$ (and on residuals if available). If a changepoint is detected (BIC/AIC/PELT), **split**.

3. **Windowing**: for slowly drifting regimes, use sliding windows in x of width h such that $| \partial_u \alpha | \ h \ll 1$ (from Sec. 3.4).

4. **Clock placebo**: multiply T by a constant; the slope $\hat{\alpha}$ must not change.

4.7 Estimation-ready dataset

Create a tidy table per bin with columns:

$$x = \log L, y = \log T, \text{family, BIN tags, replicate ID, timestamp, weights }].$$

- **Replicates.** If multiple runs at same L, aggregate to robust summaries (median y, MAD-based SE) or pass all and let ODR handle them with replicate weights.

- **Weights.** Prefer inverse-variance weights from bootstrap over simple counts.

- **Outliers.** Tag catastrophic events (hardware flags); report both **with** and **without** them.

4.8 Acceptance gates (per bin, per family)

A family contributes a slope $\hat{\alpha}_f$ **only if** all hold:

1. **Coverage:** at least 6 distinct L points and span ≥ 0.6 in $\log L$.

2. **Collapse:** regress $\tilde{y} = y - \hat{\alpha}x$ on x; require $R^2_{\text{collapse}} < 0.05$ and no visible trend (smooth check).

3. **Clock placebo:** $\hat{\alpha}$ unchanged under $T \mapsto cT$.

4. **Changepoints:** none within bin (else split and re-estimate).

5. **EIV fit quality:** ODR/TLS converged; residual diagnostics acceptable (no single leverage point dominates).

Bins or families failing any gate are flagged (`NO_COLLAPSE`, `REGIME_MIX`, `THIN_COVERAGE`, `EIV_FAIL`) and **excluded from fusion**.

4.9 Examples vs. non-examples (QC-flavored)

- **Good physical family:** L = active-qubit cluster size; T = stable calibration interval. Single firmware, stable temperature, no routing change. Collapses cleanly → accept.

- **Bad physical family:** Same, but mid-bin the PLL loop parameters change. Changepoint triggers; split required.

- **Good QEC family:** $L = d$, T = cycles to logical failure, fixed syndrome cadence. Residuals flat → accept.

- **Bad QEC family:** Mix of two cadences (fast and slow) inside one bin → kink in log–log → reject until split.

- **Good I/O family:** L = multiplexing degree; T = readout latency p95. Firmware constant; latency rises as L^α, collapse holds → accept.

- **Bad I/O family:** Switch of DSP firmware that changes per-channel overhead nonlinearly mid-bin → curvature; reject or rebin around the switch.

4.10 Summary

- We fixed **operational** (L, T) per layer and specified **instrumentation** to make them measurable.

- We defined a **binning protocol** that enforces environment constancy and guards against regime mixing.

- We set **acceptance gates** (coverage, collapse, placebo, changepoints, EIV fit) that determine whether a family's slope enters downstream fusion ($\text{ECI}_{QC}(t)$).

5. Estimation Under Errors-in-Variables (EIV) and Collapse Thresholds

We now specify **how** to estimate the binwise slope α robustly when both axes are noisy, and how to decide—via a **collapse threshold**—whether a family's data are RTM-consistent. Throughout, $x = \log L, y = \log T$. Observations are $x^{obs} = x + \xi, y^{obs} = y + \zeta$ with mean-zero errors.

5.1 Estimation targets and models

Inside a **fixed bin**, the target is the **local slope** α in

$$y = \alpha x + c + r(x),$$

with $r \equiv 0$ under exact RTM or $\mid r'(x) \mid \le \varepsilon$ under slow drift on a window. Because x is noisy, **OLS is attenuated**; we use EIV-aware estimators.

Default target: point slope α for the bin; intercept c is a **gauge** (not compared across bins).

5.2 Orthogonal Distance Regression (Total Least Squares)

Definition. ODR minimizes orthogonal residuals to a line:

$$\min_{\alpha,c} \sum_i \frac{(y_i^{obs} - \alpha x_i^{obs} - c)^2}{\sigma_y^2 + \alpha^2 \sigma_x^2}$$

with effective (possibly heterogeneous) (σ_x, σ_y) from replicate variance or bootstrap.

Practice.

- Initialize by Theil–Sen (Sec. 5.4) to avoid poor local minima.

- Use **cluster/bootstrap** (replicate or job-level) for CIs.

- If per-point SEs are available, weight them; else use robust Huber weights on orthogonal residuals.

Convergence gates.

- Condition number of the centered covariance matrix $< 10^4$.

- Jackknife leverage check: no single point contributes $> 25\%$ of slope influence.

5.3 SIMEX (when $\text{Var}(\xi)$ is known/estimated)

If you can estimate $\sigma_\xi^2 = \text{Var}(\xi)$ (e.g., repeated L at the same setting), apply **SIMEX**:

1. For $\lambda \in \Lambda = \{0.5, 1.0, 1.5, 2.0\}$, generate pseudo-samples
   $x_i^{(\lambda)} = x_i^{obs} + \sqrt{\lambda}\,\tilde{\xi}_i, \tilde{\xi}_i \sim \mathcal{N}(0, \sigma_\xi^2).$

2. Fit a naive slope $\hat{\alpha}(\lambda)$ by ODR or OLS.

3. Fit a quadratic $\hat{\alpha}(\lambda) = a + b\lambda + c\lambda^2$ and **extrapolate to $\lambda = -1$**:
   $\hat{\alpha}_{\text{SIMEX}} = a - b + c.$

Use. Prefer ODR as the base fitting routine; report SIMEX as a **sensitivity** estimate next to ODR. If σ_ξ^2 is uncertain, give a band (low/med/high) for $\hat{\alpha}_{\text{SIMEX}}$.

5.4 Theil–Sen (robust median slope)

The **Theil–Sen** slope is the median of all pairwise slopes

$$\alpha_{ij} = \frac{y_j^{obs} - y_i^{obs}}{x_j^{obs} - x_i^{obs}} (i < j),$$

with a robust intercept from the median of $y_i^{obs} - \hat{\alpha}x_i^{obs}$.

Role.

- Initialization for ODR.

- **Outlier-robust** cross-check reported alongside ODR.

- When EIV is severe and σ_ξ^2 is unknown, Theil–Sen may still be stable (expect mild attenuation).

5.5 Windowing and finite-window bias

If slow drift is suspected, estimate slopes on **symmetric windows** in x of width h. From the adiabatic bias bound,

$$\hat{\alpha}(u; h) = \alpha(u) + O(\varepsilon h),$$

choose h so that $\varepsilon h \ll 1$. Practically: start with $h \approx 0.8$ in $\log L$ span if coverage allows; shrink until collapse passes (Sec. 5.7) without exploding variance.

5.6 Uncertainty and diagnostics

- **Bootstrap** (pairs within bin or block/cluster if natural replicates exist) for 50/95% CIs.

- **Jackknife-after-bootstrap** to detect leverage points.

- **Residual plots**: orthogonal residual vs. x; LOESS smooth must be flat within bands.

- **EIV adequacy**: if OLS and ODR differ by ≥ 0.2 absolute slope **and** ODR CI excludes OLS, report EIV as material.

5.7 Collapse threshold (specification gate)

Given $\hat{\alpha}$, compute residuals $\tilde{y}_i = y_i^{obs} - \hat{\alpha} x_i^{obs} - \hat{c}$ and regress $\tilde{y}$ on x (with the same weights used in estimation). Define

$$R^2_{\text{collapse}} := R^2(\tilde{y} \sim x).$$

Decision rule (default):

- Accept the bin if **all** hold:

    1. $R^2_{\text{collapse}} < 0.05$ (or the 95% CI of the slope in $\tilde{y} \sim x$ contains 0),

    2. LOESS smooth shows no trend,

    3. **Clock placebo**: scaling $T \mapsto cT$ leaves $\hat{\alpha}$ and R^2_{collapse} unchanged,

    4. Changepoint scan (PELT/BIC) finds none inside the bin.

- Otherwise flag (`NO_COLLAPSE` or `REGIME_MIX`) and **do not** publish a slope or include it in fusion.

5.8 Coverage and leverage gates

To avoid brittle fits:

- **Distinct L points** ≥ 6 and $\log L$ span ≥ 0.6.

- **Balanced leverage:** the largest leverage point contributes $\leq 25\%$ of the ODR slope influence.

- **Replicates:** if > 3 replicates per L, either summarize to a robust mean/SE or pass replicate weights to ODR.

Bins failing these gates are flagged `THIN_COVERAGE` or `LEVERAGE_RISK`.

5.9 Putting it together (per-bin algorithm)

1. **Prep:** build the tidy table (Sec. 4.7); run changepoint scan; window if needed.

2. **Init:** compute Theil–Sen slope/intercept; remove obvious catastrophics (keep both versions for sensitivity).

3. **Fit ODR/TLS:** weighted by replicate SEs; obtain $\hat{\alpha}$, $\hat{c}$, bootstrap CIs.

4. **SIMEX (optional):** if σ_ξ^2 is available, compute $\hat{\alpha}_{\text{SIMEX}}$.

5. **Collapse gate:** compute R^2_{collapse}, smooth check, placebo clock.

6. **Decision:** if all gates pass, **accept** $\hat{\alpha}$ with uncertainty; else **reject/split**.

7. **Report:** slope, CI, diagnostics (collapse R^2, leverage plot, changepoints). Store flags.

5.10 What we publish per accepted family

- $\hat{\alpha}_f \pm 50/95\%$ CI (ODR); Theil–Sen as robustness; SIMEX band if applicable.

- Collapse diagnostics: R^2_{collapse}, placebo check, window width h.

- Coverage: # distinct L, $\log L$ span, leverage summary.

- Notes: any exclusions (catastrophics), changepoint status.

Only accepted families enter **fusion** (Sec. 6). If ≥ 2 families pass, we apply random-effects with Q, I^2 and heterogeneity gates; otherwise we report family-wise slopes without fusion.

5.11 Summary

- Use **ODR/TLS** as the primary EIV estimator; **Theil–Sen** for robust init/check; **SIMEX** when σ_ξ^2 is estimable.

- Enforce **collapse** as a **specification test** ($R^2_{\text{collapse}} < 0.05$ + placebo + no changepoints).

- Control **finite-window bias** by choosing h small enough (adiabatic regime) and splitting bins when needed.

- Publish complete **diagnostics** and **flags**; only clean families proceed to fusion and to the real-time $\text{ECI}_{\text{QC}}(t)$.

6. Building the Real-Time Indicator $\text{ECI}_{\text{QC}}(t)$

We now construct a **single, real-time** coherence indicator for a platform by fusing the **accepted** family-wise slopes $\{\hat{\alpha}_{f,t}\}$ from Section 5. The fusion is **random-effects** (to acknowledge between-family heterogeneity), runs on a rolling clock, and drives **QA gates** and **decoherence alerts**.

6.1 Inputs and preconditions (per time t)

For each family $f \in \mathcal{F}_t$ (Physical, QEC, Compiler/Runtime, I/O–Cryo):

- A binwise estimate $\hat{\alpha}_{f,t}$ with variance $\hat{\sigma}_{f,t}^2$ (bootstrap or replicate-weighted),

- Collapse passed (Section 5.7), coverage/leverage gates satisfied (Section 5.8),

- Environment tags (BIN) unchanged within the window that produced $\hat{\alpha}_{f,t}$.

A fusion at time t proceeds **only if** $| \mathcal{F}_t | \geq 2$.

6.2 Random-effects fusion

We estimate the between-family variance $\hat{\tau}_t^2$ (default **REML**; DerSimonian–Laird as sensitivity). Define weights

$$w_{f,t} = \frac{1}{\hat{\sigma}_{f,t}^2 + \hat{\tau}_t^2}.$$

Then the fused slope and its variance are

$$\hat{\alpha}_{\text{QC}}(t) = \frac{\sum_{f \in \mathcal{F}_t} w_{f,t}\, \hat{\alpha}_{f,t}}{\sum_{f \in \mathcal{F}_t} w_{f,t}}, \qquad \text{Var}\left(\hat{\alpha}_{\text{QC}}(t)\right) = \frac{1}{\sum_{f \in \mathcal{F}_t} w_{f,t}}.$$

Report 50% and 95% intervals via normal approximation or by a **bootstrap-over-families** (resample families with replacement, recompute $\hat{\tau}_t^2$ and the fused mean).

6.3 Heterogeneity diagnostics and gates

Compute the fixed-effect baseline

$$w_{f,t}^{FE} = \frac{1}{\hat{\sigma}_{f,t}^2}, \qquad \hat{\alpha}_{FE}(t) = \frac{\sum_f w_{f,t}^{FE}\, \hat{\alpha}_{f,t}}{\sum_f w_{f,t}^{FE}}.$$

Cochran's Q and I^2:

$$Q_t = \sum_f w_{f,t}^{FE} (\hat{\alpha}_{f,t} - \hat{\alpha}_{FE}(t))^2, \qquad I_t^2 = \max\left\{0, \frac{Q_t - (|\mathcal{F}_t| - 1)}{Q_t}\right\} \times 100\%.$$

Fusion gates (pre-registered):

- Proceed with a single number **only if**
  (i) $|\mathcal{F}_t| \geq 2$,
  (ii) $I_t^2 < 50\%$ *(moderate or lower heterogeneity)*, and
  (iii) REML converges with finite $\hat{\tau}_t^2$ not exceeding a historical cap (e.g., $\leq$ 90th percentile over past clean windows).

- If any fails, **withhold fusion** and publish family-wise $\hat{\alpha}_{f,t}$+ diagnostics; flag FAMILY_DIVERGENCE.

6.4 Real-time operation (rolling windows)

- **Cadence.** Recompute each family's $\hat{\alpha}_{f,t}$ on a **rolling window** in $x = \log L$ of width h (chosen by the adiabatic rule; Sec. 5.5) and a **wall-clock horizon** (e.g., last 7–28 days of data).

- **Backfill and missingness.** If a family is missing at t, fuse over the available $\mathcal{F}_t$ provided $|\mathcal{F}_t| \geq 2$; otherwise **suspend** $\text{ECI}_{\text{QC}}(t)$ and publish a `THIN_FAMILIES` flag.

- **Clock placebo.** Once per day, multiply all contributing T by a constant and verify $\hat{\alpha}_{\text{QC}}(t)$ and I_t^2 are unchanged (stored as a QA artifact).

6.5 Decoherence events (alerting logic)

We define a **decoherence event** as a significant, QA-clean **drop** in $\text{ECI}_{\text{QC}}(t)$, robust to smoothing and not explained by heterogeneity spikes.

Filters:

1. **Smoothing:** maintain a 3-point median $\tilde{\alpha}(t)$ of $\hat{\alpha}_{\text{QC}}(t)$.

2. **Z-score:** $Z(t) = \frac{\tilde{\alpha}(t) - \text{EWMA}_{30}[\tilde{\alpha}]}{\hat{\sigma}_{EWMA}(t)}$.

Alert tiers (default):

- **Advisory:** $Z(t) \leq -1.5$ for ≥2 consecutive ticks **and** $I_t^2 < 50\%$.

- **Watch:** $Z(t) \leq -2.0$ once **or** persistent $Z(t) \leq -1.5$ for ≥4 ticks, $I_t^2 < 40\%$.

- **Warning:** $Z(t) \leq -2.5$ and a coincident family-wise drop (≥2 families with $Z_f \leq -2$).

Playbooks triggered: throttle scheduling (reduce concurrency/multiplexing), run segmented recalibration, or switch to RTM-aware routing until $\tilde{\alpha}(t)$ normalizes.

6.6 Reporting and visualization

- **Primary panel:** $\hat{\alpha}_{\text{QC}}(t)$ with 50/95% bands, heterogeneity ribbon colored by I_t^2 (green <25%, amber 25–50%, red ≥50%).

- **Forest plot:** per-family $\hat{\alpha}_{f,t}$, weights $w_{f,t}$, and CIs; show $Q_t, I_t^2, \hat{\tau}_t^2$.

- **Collapse dashboard:** per family, show R^2_{collapse}, LOESS residuals, window width h, coverage and leverage metrics.

- **Flags legend:** NO_COLLAPSE, REGIME_MIX, LEVERAGE_RISK, THIN_COVERAGE, FAMILY_DIVERGENCE, THIN_FAMILIES.

6.7 Sensitivity and ablation

- Publish the **fixed-effect** summary $\hat{\alpha}_{FE}(t)$ alongside random-effects.

- Report DL-based $\hat{\tau}_{DL}^2$ as a sensitivity.

- **Leave-one-family-out**: recompute $\hat{\alpha}_{QC}^{(-f)}(t)$ to expose dominance.

- **Clock placebos** and **shuffle nulls** (shuffle L within family) must not produce tiered alerts; if they do, review gates.

6.8 Governance and provenance

Every fused point stores:

- Source families and their BIN tags,

- Estimator settings (ODR init, bootstrap seeds, h),

- Collapse metrics, $Q_t, I_t^2, \hat{\tau}_t^2$,

- Placebo outcome hashes,

- Versioned code/config (methods YAML).

This ensures **reproducibility** and enables post-mortems when alerts fire.

6.9 Summary

$\text{ECI}_{QC}(t)$ is a **random-effects fusion** of QA-clean, binwise slopes. Heterogeneity gates ($I_t^2 <$ 50%, | $\mathcal{F}_t$ |≥ 2) prevent misleading single numbers when proxies disagree. Real-time smoothing and Z-scores turn slope dynamics into **actionable alerts** for **decoherence events**, while dashboards and provenance keep the system auditable.

7. RTM-Aware Design: Engineering α without Sacrificing Throughput

This section turns RTM into **design levers**. Goal: increase the **coherence exponent** α (stronger tempo stratification across scale) while keeping or improving throughput. We give layer-specific controls, optimization targets, and guardrails.

7.1 Design objective and guardrails

We treat α as an **operational objective** within a bin:

$$\max_{\text{controls } \theta} \ \alpha(\theta) \ \ s.t. \ \ throughput \ \geq \ B, \ \ fidelity \ \geq \ F, \ \ collapse \ passes.$$

- **Controls** θ: scheduler parameters, QEC cadence/jitter, routing constraints, multiplexing limits, module sizes.

- **Constraints**: a throughput floor $\mathcal{B}$ (e.g., jobs/hour), fidelity floor $\mathcal{F}$, and **collapse gates** (Sec. 5.7).

- **Monitor**: track per-family $\hat{\alpha}_f$ and the fused $\hat{\alpha}_{\text{QC}}(t)$ with QA (Sec. 6).

7.2 Scheduler: batching & variance-aware routing

Problem. Long, tightly coupled operations launched in parallel **flatten** α (fast cascades across scale).

Controls.

1. **Wavefront batching (readout & long ops).** Partition time into short waves; pack readouts into waves instead of free-running concurrency.

2. **Staggered resets.** Add small offsets $\delta \in [-\epsilon, \epsilon]$ to reset times to avoid synch peaks.

3. **Low-variance routing.** Prefer routes with **low path-time variance** even if path length increases slightly.

Objective. For a job DAG with ops o having nominal durations τ_o and routes $p(o)$:

$$\min_{\text{schedule}, p(\cdot)} \ \underbrace{\text{Var}_t[N_{\text{long}}(t)]}_{\text{desynchronize heavy ops}} \ + \ \lambda \sum_{o \in \mathcal{O}} \underbrace{\text{Var}(T_{\text{route}}(p(o)))}_{\text{low-variance routing}}.$$

subject to makespan budget. This reduces temporal "pile-ups," lifting α.

Heuristic (greedy, practical).

- Sort ops by duration desc; assign start times into **waves** so that each wave's total long-op load is balanced.

- For each route candidate, penalize time-variance and crosstalk score; pick minimum penalized cost.

7.3 QEC cadence: avoid phase-lock (jitter/desynchronization)

Problem. A fixed syndrome cadence can **phase-lock** with physical noise rhythms, creating cross-layer synchronization → α_{QEC} falls.

Controls.

- **Micro-jitter** the cycle period: $P_k = P\,(1 + \eta_k)$ with $\eta_k \sim \mathcal{U}[-\rho, \rho]$, $\rho \ll 1$ (e.g., 1–3%).

- **Multi-phase extraction:** split the code into sublattices whose cycles are offset by small phases ϕ_j.

Design rule. Choose ρ so that the **main lobe** of the syndrome cycle's line spectrum moves **off** strong peaks of the error PSD while keeping decoder timing valid. Validate by: (i) increased $\hat{\alpha}_{\mathrm{QEC}}$ vs. d, (ii) stable logical error at fixed d.

7.4 Gradients and wells of α

Two architectural motifs to **steer flows**:

- **Gradient:** arrange resources so α **increases** towards critical compute regions. Small disturbances decay as they travel inward.

- **Well:** create a **high-α basin** around sensitive qubits (e.g., clocking and buffering that slow large-scale cascades).

Implementation cues. Increase temporal buffering (queues, damped scheduling) and reduce crosstalk fanout as you approach the "core," but cap buffering (Sec. 7.1 guardrails) so throughput doesn't suffer.

7.5 Modular sizing: pick a sweet spot by balancing intra vs. inter latency

Let total qubits Q be partitioned into Q/m modules of size m. Approximate **characteristic time**:

$$T(m) = A m^a + B \left(\frac{Q}{m}\right)^b \quad \text{(intra-module cost + interconnect cost)}.$$

Optimal module size (minimizes T):

$$m^\star = \left(\frac{B\,b}{A\,a}\right)^{\frac{1}{a+b}} Q^{\frac{b}{a+b}}.$$

- $a > 0$: intra-module scaling (e.g., calibration, routing within module).

- $b > 0$: inter-module scaling (e.g., photonic/ion link latency).

Design use. Measure a, b empirically (RTM per mechanism), estimate A, B, compute $m^\star$. Operate near $m^\star$ and verify that $\hat{\alpha}$ **does not collapse** (still power-like) in that neighborhood.

7.6 Multiplexing & I/O: hold tails in check

Problem. Aggressive multiplexing reduces per-shot time, but can synchronize queue tails $\rightarrow \alpha_{IO} \downarrow$.

Controls.

- Cap multiplexing such that the **tail ratio** $p95/p50$ of readout latency stays below a threshold (e.g., ≤ 1.6).

- Use **phase-offset readout windows** across channels to avoid coherent tail growth.

- Buffer sizing: maintain buffer utilization $< 70\%$ to avoid tail amplification.

Signal. If $p95/p50$ grows and $\hat{\alpha}_{IO}$ drops with clean collapse, back off multiplexing and introduce offsets.

7.7 Online control loop (closed-loop α engineering)

A simple controller to keep α high under constraints:

```
every Δt:

  estimate {α_f(t), σ_f(t)} per accepted family (Sec. 5)

  if |F_t| ≥ 2 and I^2_t < 50%:

      compute α_QC(t) (Sec. 6)
```

```
if α_QC(t) < α_floor and constraints met:

    apply actions A = {↑wave size, ↑reset jitter ρ, ↑routing penalty on variance,

                      ↓multiplex cap, move toward m*}

else if throughput < B:

    relax A minimally (keep collapse passing)

log QA: collapse R^2, I^2_t, flags; revert actions if flags trip
```

- α_{floor}: pre-registered minimal acceptable fused slope.

- **Revert** any action that causes `NO_COLLAPSE` or $I_t^2 \geq 50\%$.

7.8 Safety and validation

- Any intervention must **re-pass collapse** in the affected families.

- Run A/B windows (≥2–4 weeks) with **pre-registered** KPIs: throughput, makespan, logical error, uptime, $p95/p50$, and $\hat{\alpha}_f$.

- If α rises but KPIs worsen beyond budgets, you are **over-layering** (too much buffering). Roll back to the Pareto frontier.

7.9 Quick-start playbooks

- **If α_{QEC} ↓:** add 1–3% cadence jitter; introduce 2–3 phase groups for syndrome; re-measure collapse.

- **If α_{IO} ↓:** reduce multiplex cap 10–20%; add 1–2 cycle offsets; keep $p95/p50 \leq 1.6$.

- **If α_{runtime} ↓:** enable readout batching; penalize high-variance routes; cap concurrent long ops per wave.

- **Architectural planning:** estimate a, b, A, B and set module size near $m^\star$; confirm power-like scaling around that point.

7.10 Summary

- **Scheduler** (waves, staggered resets, low-variance routing) and **QEC cadence** (micro-jitter, multi-phase) are first-line levers to **raise α**.

- **Modular sizing** admits a closed-form optimum $m^\star$ balancing intra/inter costs; operate near it while watching collapse.

- **I/O controls** keep latency tails from synchronizing.

- A **closed-loop controller** maintains α above a floor under throughput/fidelity budgets.

8. Falsifiable Experimental Protocols (Superconducting & Trapped-Ion)

This section specifies **testable** RTM-QC experiments with concrete (L, T) choices, data collection, analysis plans, and success criteria. Each protocol is binwise (fixed environment) and includes **placebos**, **changepoint guards**, and a **pre-registered** decision table.

8.1 Common scaffolding (applies to all protocols)

BIN (environment) lock.
{platform; temperature band; firmware hash (FPGA/DSP); topology ID; routing policy; syndrome cadence; utilization band}. Any change $\Rightarrow$ new bin.

Data schema (tidy). For each record:

$$x = \log L, y = \log T, \text{family, BIN tags, replicate ID, timestamp, weights]}$$

QA gates (must pass):

- Coverage: ≥6 distinct L, span ≥0.6 in $\log L$.

- EIV fit converged (ODR), leverage <25%, robust init (Theil–Sen).

- Collapse: $R^2_{\text{collapse}} < 0.05$, no LOESS trend, clock placebo holds.

- Changepoints: none inside bin (else split).

Outcomes (primary, per family):

- Slope $\hat{\alpha}_f$ with 50/95% CI; collapse diagnostics.

- For fused results, $\hat{\alpha}_{\text{QC}}(t), Q, I^2, \hat{\tau}^2$ (Sec. 6).

Statistical plan.
Bootstrap CIs (pairs/cluster). Predefine **minimal detectable effect** (MDE) on α (e.g., $\Delta\alpha =$

0.15) and **operational KPIs** (throughput, makespan, logical error rate, uptime, p95/p50). Thresholds below.

8.2 Protocol A — Physical layer (Superconducting)

Hypothesis (H1-Phys). Increasing **cluster desynchronization** (staggered resets + readout waves) **raises** α_{phys} without exceeding throughput budget.

Design.

- L: active-qubit **cluster size** (simultaneously engaged).

- T: **stable calibration interval** (time to first out-of-tolerance flag in cluster).

- Arms: **Control** (baseline scheduler) vs. **RTM-aware** (readout batching + staggered resets, ±2–4% offsets).

- Duration: 2–4 weeks; interleave arms daily to balance drift.

Analysis.

- Fit ODR per arm, pass collapse.

- Primary effect: $\Delta\hat{\alpha}_{phys} = \hat{\alpha}_{RTM} - \hat{\alpha}_{CTRL}$.

- KPI guardrails: throughput drop ≤5%, no increase in gate/RO error >0.2σ.

Success criteria.

- $\Delta\hat{\alpha}_{phys} \geq 0.15$ and CI excludes 0, **and** guardrails satisfied.

- If collapse fails in any arm, declare **inconclusive** and rebin.

Placebos. Multiply T by a constant; $\hat{\alpha}$ unchanged. Shuffle L within day; no significant slope.

8.3 Protocol B — QEC cadence (Superconducting or Ions)

Hypothesis (H1-QEC). Introducing **micro-jitter** (1–3%) in syndrome period and/or **multi-phase extraction** increases α_{QEC} vs. code distance d at fixed decoder.

Design.

- L: **code distance** d (e.g., $d \in \{3,5,7,9\}$).

- T: **cycles to logical failure** (median or survival quantile at fixed target error).

- Arms: Control (fixed period P) vs. Jitter ($P_k = P(1 + \eta_k), \eta_k \sim \mathcal{U}[-0.02, 0.02]$) and/or 2–3 **phase groups**.

- Keep decoder parameters fixed; no change in noise bias mitigation.

Analysis.

- ODR per arm; collapse gate.

- Effect: $\Delta\hat{\alpha}_{\text{QEC}}$.

- KPI guardrails: logical error at fixed d not worse by >5% relative.

Success criteria.

- $\Delta\hat{\alpha}_{\text{QEC}} \geq 0.15$ with CI excluding 0 and guardrails pass.

Diagnostics. Check PSD of error processes; confirm jitter moves cadence lines off dominant peaks.

8.4 Protocol C — compiler/runtime scheduling

Hypothesis (H2-Run). Wavefront batching of readout and **low-variance routing** reduce synchronization cascades, increasing α_{runtime} and lowering makespan tails.

Design.

- L: **post-mapping circuit width** (or active layers).

- T: **makespan** (submit→complete).

- Arms: Baseline policy vs. RTM-aware (waves + variance-penalized routing).

- Control utilization band; same job mix across arms.

Analysis.

- ODR slope per arm; collapse.

- KPIs: median makespan (≤ baseline), p95/p50 latency ↓ ≥10%.

Success criteria.

- $\Delta\hat{\alpha}_{\text{runtime}} \geq 0.10$(CI excludes 0) and p95/p50 improves ≥10%.

8.5 Protocol D — I/O–Cryo multiplexing

Hypothesis (H2-IO). Phase-offset readout windows across channels maintain or raise α_{IO} while reducing p95 tails at a given multiplexing degree.

Design.

- L: **multiplexing degree** (channels/line).

- T: **readout latency p95** (and p50).

- Arms: Synchronous windows vs. offset windows (phase pattern ϕ_j).

- Sweep L across operational range.

Analysis & success.

- $\Delta\hat{\alpha}_{IO} \geq 0.10$; p95/p50 ≤ 1.6 in RTM arm over majority of L; collapse passes.

8.6 Protocol E — Modular sizing (planning study)

Hypothesis (H3-Mod). There exists a module size $m^\star$ that minimizes $T(m) = Am^a + B(Q/m)^b$ with empirically measured $a, b > 0$, and operating near $m^\star$ preserves power-like scaling (collapse holds).

Design.

- Platforms with photonic/ion links between modules.

- Measure $T(m)$ by varying module size (or emulating interconnect cost) at fixed total Q.

- Fit a, b, A, B via ODR on each term's dataset; compute $m^\star$.

Success criteria.

- Observed $T(m)$ minimized near $m^\star$ (within CI), and log–log fits around $m^\star$ retain collapse (no curvature).

8.7 Fusion and alerting (cross-protocol)

Across A–D, if ≥2 families pass gates at overlapping times, compute $\hat{\alpha}_{QC}(t)$ (Sec. 6).
H2 (anticipation): declare a **decoherence event** if Z-score tiers (Sec. 6.5) are met; test **lead–lag** vs. spikes in logical error/makespan/queues. Additive predictive value is assessed against baselines (fidelity, utilization, temperature) using time series regression with HAC errors; pre-register horizons (e.g., 7–30–90 days).

8.8 Placebos, shuffles, and robustness

- **Clock placebos:** multiply all T by constants; $\hat{\alpha}$ and R^2_{collapse} invariant.

- **Shuffle nulls:** permute L within day; slopes collapse to ~ 0 (within CI).

- **Leave-one-family-out** fusion to reveal dominance.

- **Changepoints**: automatic split if detected; re-estimate on both sides.

8.9 Power and duration (rules of thumb)

- With span ≥ 0.8 in $\log L$, 8–12 distinct L points, and moderate noise (SNR$\approx$5–10), ODR detects $\Delta\alpha \approx 0.10$–$0.15$ at 95% with $\approx$200–400 total observations per arm.

- If noise is higher or drift suspected, shrink windows (Sec. 5.5) and extend duration.

8.10 Decision table (pre-registered)

Outcome	Action
$\Delta\hat{\alpha} \geq$ MDE **and** guardrails pass	Promote intervention to production in that bin; monitor with $\text{ECI}_{\text{QC}}(t)$.
$\Delta\hat{\alpha}$ significant but KPI guardrail violated	Tune intensity (e.g., reduce buffering/jitter) and retest.
Collapse fails or heterogeneity high ($I^2 \geq$ 50%)	Do not fuse; report family-wise; revisit binning or mechanisms.
No effect (($\Delta\hat{\alpha} \approx 0$)	Document as *scope boundary*; keep as negative control.

8.11 Ethics, safety, and reproducibility

- **Safety:** no unsafe RF power increase; jitter bounds keep decoders valid; rollback on `NO_COLLAPSE` or KPI breach.

- **Reproducibility:** versioned methods YAML (BIN, estimator settings, seeds), public plots (collapse panels, forest plots), and stored placebo/shuffle artifacts.

- **Transparency:** publish both successes and failures (negative results define scope).

8.12 Summary

These protocols make RTM-QC **falsifiable**: each claims a directional change in α from a specific control, under binwise constancy, with collapse as a specification test and operational guardrails. Success improves not only the slope but also **run-time stability** (tails, recalibrations) without sacrificing throughput.

9. Results Templates and Reporting Standards

This section defines **what to publish** once the protocols (Sec. 8) are run. It standardizes figures, tables, robustness panels, and a one-page checklist so that results are interpretable, reproducible, and directly comparable across labs and platforms.

9.1 Figure set (minimum)

Fig. 1 — Collapse panels (per accepted family).
Four small multiples per family f within a bin:

1. **Log–log fit:** $y = \log T$ vs. $x = \log L$ with ODR line and 95% band.

2. **Residual vs. x:** $\tilde{y} = y - \hat{\alpha}x - \hat{c}$ with LOESS; show R^2_{collapse}.

3. **Coverage/leverage:** scatter highlighting leverage points; annotate span in $\log L$, # distinct L.

4. **Placebo check:** overlay of fits before/after $T \mapsto cT$ (curves coincide).

Fig. 2 — Forest plot & heterogeneity.
Per time slice (or per experiment arm), show $\hat{\alpha}_f \pm$CI, weights w_f, the fused $\hat{\alpha}_{\text{QC}}$ (diamond), and heterogeneity stats: $Q, I^2, \hat{\tau}^2$.

Fig. 3 —ECI_{QC}(t) time series.
Rolling fused slope with 50/95% bands; background ribbon colored by I^2 (green <25%, amber 25–50%, red ≥50%). Mark **decoherence events** (advisory/watch/warning) and platform events (recalibrations, firmware changes).

Fig. 4 — KPI panel (paired with Fig. 3).
Aligned time axes for: logical error rate (at fixed d), makespan median and p95, queue p95, uptime between recalibrations. Overlay shaded regions for alert tiers from Fig. 3.

Fig. 5 — A/B outcomes (per protocol).
For each arm: distribution plots (violin/box) of $\hat{\alpha}_f$, makespan p95/p50, logical error; include $\Delta\hat{\alpha}$ with CI and guardrails.

Optional Fig. 6 — Spectral diagnostics (QEC).
PSD of error processes showing how cadence jitter/multi-phase moves line spectra off dominant peaks.

9.2 Core tables

Table 1 — Accepted families (per bin/arm).

Family	#L pts	$\log L$ span	$\hat{\alpha}_f$ (ODR, 50/95% CI)	Theil–Sen	SIMEX band	(R^2_{coll})	Leverage max	Flags
Physical	9	1.05	0.62 [0.55, 0.70]	0.60	0.58–0.66	0.02	0.18	—
QEC	8	0.82	0.74 [0.66, 0.82]	0.71	—	0.03	0.22	—
…	…	…	…	…	…	…	…	…

Table 2 — Fusion & heterogeneity (per time slice or arm).

Time/Arm	Families	$\hat{\alpha}_{QC} \pm$ SE	(Q) (df)	I^2	$\hat{\tau}^2$	Fusion?
RTM-aware	3	0.69 ± 0.04	3.2 (2)	37%	0.005	Yes
Control	3	0.54 ± 0.05	6.8 (2)	71%	0.018	**No** (report family-wise)

Table 3 — Protocol outcomes (A/B).

Protocol	Metric	Control	RTM-aware	Effect (Δ)	95% CI	Pass guardrail?
A (Phys)	$\hat{\alpha}_{phys}$	0.48	0.64	+0.16	[0.07, 0.25]	✓
A (Phys)	Throughput	100%	97%	–3%	[–6, 0]%	✓
B (QEC)	$\hat{\alpha}_{QEC}$	0.68	0.83	+0.15	[0.06, 0.24]	✓
C (Run)	p95/p50	1.85	1.60	–0.25	[–0.35, –0.15]	✓

Table 4 — Pre-registered thresholds & flags.

Gate	Threshold	Status
Collapse R^2	< 0.05	Pass
Heterogeneity I^2	< 50% for fusion	Pass
MDE on $\Delta\alpha$	≥ 0.10–0.15	Pass
KPI guardrails	≤5% throughput loss; ≤+5% logical error	Pass

9.3 Robustness and sensitivity panel

- **Estimators:** ODR (primary), Theil–Sen, SIMEX (± bands for σ_ξ^2).

- **Windows:** repeat with $h\pm$ 25%; $\hat{\alpha}$ stable and collapse still passing.

- **Placebos:** clock rescaling; **Shuffles:** permute L within-day—slope $\rightarrow \sim 0$.

- **Leave-one-family-out fusion:** report $\hat{\alpha}_{QC}^{(-f)}$.

- **Catastrophics:** re-estimate excluding flagged events; show Δ.

- **Fixed-effect vs. random-effects:** publish both; divergence implies genuine heterogeneity.

9.4 Negative results & scope boundaries

Publish bins/arms that **failed**:

- `NO_COLLAPSE` (curvature), `REGIME_MIX` (kinks), `THIN_COVERAGE`, `LEVERAGE_RISK`, `FAMILY_DIVERGENCE` (high I^2).
  Include a short note: suspected mechanism and next steps (rebin, instrumentation change, mechanism isolation). Negative results define **where RTM does not apply**.

9.5 One-page checklist (for each figure/table set)

- ✓ BIN keys listed and unchanged.

- ✓ # distinct $L \geq 6$ and span ≥ 0.6.

- ✓ ODR converged; Theil–Sen reported; SIMEX (if σ_ξ^2 known).

- ✓ Collapse: $R^2 < 0.05$; placebo OK; no changepoints.

- ✓ Fusion: $|\mathcal{F}_t| \geq 2$; $I^2 < 50\%$; REML converged.

- ✓ KPIs: throughput, makespan p95/p50, logical error, uptime—guardrails applied.

- ✓ Robustness panel completed (windows, shuffles, LOO).

- ✓ Provenance hashes (methods YAML, seeds, code version) included.

9.6 Narrative template (short "Results" text)

Physical layer. Across 9 cluster sizes (span 1.05 in $\log L$), RTM-aware scheduling increased the slope from 0.48to 0.64 ($\Delta = 0.16$, 95% CI [0.07,0.25]); residuals showed $R^2_{\text{collapse}} = 0.02$. Throughput remained within the 5% guardrail.

QEC. With 1–3% cadence jitter, α_{QEC} rose from 0.68 to 0.83 ($\Delta = 0.15$, CI [0.06,0.24]), logical error at fixed d did not worsen.

Runtime. Wavefront batching and variance-aware routing reduced p95/p50 from 1.85 to 1.60; α_{runtime} increased by 0.12.

Fusion. Three families passed gates; $I^2 = 37\%$. The fused $\hat{\alpha}_{\text{QC}} = 0.69 \pm 0.04$. A **watch**-level decoherence alert fired on day 17; it preceded a makespan spike by 3 days.

9.7 Summary

The templates above ensure every claim is backed by: (i) **collapse** visual and numeric proof, (ii) EIV-aware estimation, (iii) **heterogeneity** accounting for fusion, (iv) KPI guardrails, and (v) complete **robustness** evidence.

10. Discussion

This section interprets RTM-QC results, clarifies how a **slope-first** view complements fidelity/QEC paradigms, and lays out trade-offs, risks, and adoption paths.

10.1 What does a higher α actually buy?

A larger binwise slope α means **time stretches more steeply with scale**, i.e., larger aggregates slow down *relative* to smaller ones within a stable environment. Operationally:

- **Shock damping:** disturbances at small scale are less likely to synchronize larger layers (runtime → QEC → I/O), reducing cascades that inflate tails (p95/p50), queues, and forced recalibrations.

- **Predictability:** higher α typically reduces **run-to-run variance** (narrower KPI distributions) because the stack's "tempo gradient" prevents alignment of rare long events.

- **Control leverage:** α is unit-agnostic; we can optimize it with scheduler/QEC/interconnect knobs without conflating unit changes (clocks) with structural change.

Not a substitute for fidelity. RTM improves **how** timing behaves across scale; it does not increase single/two-qubit fidelities by itself. Gains arrive through fewer cascades and better use of existing fidelity.

10.2 Complementarity with QEC and compilation

- **QEC:** Traditional design picks code distance d from error rates. RTM adds a second axis: **cadence geometry**. Slight **desynchronization** (jitter/multi-phase) can raise α_{QEC} at fixed d and decoder, often improving stability without extra overhead.

- **Compilation/runtime:** State-of-the-art routing minimizes depth/length. RTM asks also to minimize **time-variance** and **coincidence of long ops**, which can improve tails even if mean depth changes marginally.

10.3 Trade-offs and Pareto front

- **Throughput vs. layering:** Raising α by adding buffers/batching can reduce raw concurrency. We therefore optimize on a **Pareto front** (Sec. 7.1): increase α *subject to* throughput/fidelity floors.

- **Jitter vs. decoder timing:** Micro-jitter must stay within decoder validity; otherwise you trade higher α for logical failures.

- **Modular size:** Operating near $m^\star$ (Sec. 7.5) balances intra/inter costs, but drifting too far (bigger or smaller modules) can either flatten α (synchronization) or throttle bandwidth.

10.4 Failure modes (informative by design)

RTM's **collapse** gate turns failures into diagnostics:

- `NO_COLLAPSE`**:** curved log–log → missing mechanism (e.g., scale-dependent "clock" or nonlinear overhead).

- `REGIME_MIX`**:** kinks → hidden seams (firmware/scheduler swaps); rebin or split.

- **High** I^2**:** proxies disagree → do **not** fuse; inspect per-family controls.

Publishing these cases maps **scope boundaries** (where RTM does *not* apply), which is scientifically useful and prevents overreach.

10.5 Why a single fused indicator—and when not to use it

Pros: $\text{ECI}_{\text{QC}}(t)$ summarizes multiscale coherence, enabling **alerts** (Sec. 6.5) and trend tracking.
Cons: Fusion can hide heterogeneity. Hence the **gates** (at least two families, $I^2 < 50\%$, REML convergence). If they fail, publish **family-wise** $\hat{\alpha}_f$ only; the lack of fusion is itself a result ("the stack is speaking with different slopes").

10.6 Relation to time-changed diffusions and queueing

The PDE view (RTM as a **state-dependent clock**) explains why **tails** shrink when α rises: the **effective dynamic exponent** z increases, and exit/first-passage times scale more steeply with "radius" (Sec. 6 of the math paper). In queueing terms, scheduling that raises α **decorrelates** service bursts and dampens tail amplification.

10.7 External validity and portability

Because α is **gauge-invariant**, comparisons hold across labs and generations when bins are matched (environment keys). The same pipeline ports to **trapped ions**, **superconducting**, **neutral atoms**, and **annealers** with layer-appropriate (L, T). What changes is instrumentation; the **collapse logic** and **EIV estimation** remain.

10.8 Adoption path (practical)

1. **Shadow mode:** compute per-family $\hat{\alpha}_f$ and collapse panels without changing operations.

2. **Low-risk knobs:** enable **readout batching**, **staggered resets**, and tiny **cadence jitter** (≤3%).

3. **Close the loop:** bring $\text{ECI}_{\text{QC}}(t)$ into on-call dashboards with alert tiers and playbooks.

4. **Architectural planning:** measure a, b, A, B (Sec. 7.5) to choose module sizes; iterate quarterly.

10.9 Open questions

- **Decoder co-design:** how to include α directly in decoders' scheduling/graph updates?

- **Learning controllers:** can RL tune α subject to KPI floors without violating collapse?

- **Holonomy tests:** practical statistics to distinguish curvature from topological obstructions (global collapse failure).

- **Cross-layer causality:** when do α changes at the physical layer *cause* changes at runtime vs. just correlate via utilization?

10.10 Takeaway

RTM-QC adds a **third axis**—the **geometry of tempo**—to fidelity and scale. With strict gates (collapse, heterogeneity) and modest controls (batching, jitter, routing variance), α becomes a reliable lever for **stability and throughput**, yielding early warnings and design guidance while respecting scientific falsifiability.

11. Limitations & Scope

Bin dependence. RTM is a **binwise** theory. If the environment (temperature, firmware, topology, decoder, utilization) drifts, the slope α is undefined until the bin is split. Results are only valid within clearly documented BIN keys.

Proxy choice sensitivity. (L, T) proxies must reflect a **single dominant mechanism** per family. Mis-specified proxies (e.g., mixing readout and routing in the same T) induce curvature and validly fail collapse.

Finite-window bias. When $\alpha(u)$ drifts, any finite window of width h incurs $O(\varepsilon h)$ bias. Our adiabatic guidance mitigates but does not eliminate it; reported $\hat{\alpha}$ should be interpreted as **local**.

EIV model assumptions. ODR/TLS and SIMEX assume well-behaved errors (mean-zero, finite moments) and independence from x. Heavy-tailed or state-dependent errors require robustness checks (Theil–Sen, bootstrap, sensitivity bands).

Fusion heterogeneity. Random-effects fusion is appropriate only when families are **commensurate** and $I^2 < 50\%$. Otherwise the single-number indicator is withheld by design; RTM does not force agreement across mechanisms.

Causality limits. α is **structural but not causal** by default. Design sections propose interventions and A/B protocols, yet causal claims require the pre-registered controls and guardrails we specify.

Scope boundaries. Systems with **non-power** timing (persistent curvature), **scale-dependent clocks** (overheads that grow with L inside a bin), or strong **holonomy** (global seams) lie **outside** RTM's applicability. In such domains, treat α as undefined and publish negative results.

12. Methods & Reproducibility

12.1 Data schema and BINs

- **BIN key:** {platform, temperature band, firmware hash (FPGA/DSP), topology ID, routing policy, syndrome cadence, utilization band}.

- **Tidy table (per bin):** [x=log L, y=log T, family, BIN tags, replicate_id, timestamp, weight].

- **Coverage gates:** ≥6 distinct L, span ≥0.6 in $\log L$.

12.2 Estimation pipeline (per family, per bin)

1. **Changepoint scan:** PELT/BIC on (x, y) and on residuals if available; split if detected.

2. **Init:** Theil–Sen slope/intercept; flag catastrophics; build replicate weights.

3. **Primary fit:** ODR/TLS (orthogonal residuals) with replicate or bootstrap SEs.

4. **SIMEX (optional):** when σ_ξ^2 is estimable; extrapolate to $\lambda = -1$.

5. **Collapse test:** regress $\tilde{y} = y - \hat{\alpha}x - \hat{c}$ on x; require $R^2_{\text{collapse}} < 0.05$, flat LOESS, clock placebo holds.

6. **Diagnostics:** leverage ≤25%; residual plots; window width h logged.

7. **Accept/Reject:** accept if all gates pass; else flag (`NO_COLLAPSE`, `REGIME_MIX`, `THIN_COVERAGE`, `LEVERAGE_RISK`, `EIV_FAIL`).

12.3 Fusion and heterogeneity (rolling)

- **Weights:** $w_f = 1/(\hat{\sigma}_f^2 + \hat{\tau}^2)$ with $\hat{\tau}^2$ via REML (DL as sensitivity).

- **Fused slope:** $\hat{\alpha}_{\text{QC}} = \sum w_f \hat{\alpha}_f / \sum w_f$; **variance:** $1/\sum w_f$.

- **Diagnostics:** fixed-effect baseline, **Cochran's** Q and I^2.

- **Gates:** fuse only if $|\mathcal{F}| \geq 2$ and $I^2 < 50\%$. Otherwise publish family-wise.

12.4 Real-time operation and alerts

- **Rolling windows:** sliding horizon in x (width h) and wall-clock (7–28 days).

- **Smoothing:** 3-point median; **Z-score** against 30-day EWMA.

- **Alert tiers:** Advisory/Watch/Warning thresholds (Sec. 6.5).

- **Playbooks:** throttle concurrency, stagger resets, cadence jitter, variance-aware routing; all interventions must re-pass **collapse**.

12.5 Robustness & sensitivity

- **Estimators:** publish ODR (primary), Theil–Sen, SIMEX bands.

- **Windows:** ±25% h sensitivity; $\hat{\alpha}$ stability required.

- **Placebos & shuffles:** clock rescaling invariance; L-shuffles yield near-zero slopes.

- **Leave-one-family-out** fusion; **fixed-effect** vs **random-effects** comparison.

12.6 Provenance (methods YAML)

- BIN keys, estimator settings, bootstrap seeds, SIMEX Λ, window h, collapse thresholds, heterogeneity gates, versions of analysis code.

- All plots and numbers include hash of the methods YAML; re-runs with the same YAML reproduce numbers within bootstrap noise.

13. Conclusion & Outlook

We presented **RTM-aware quantum computing (RTM-QC)**: a **slope-first** framework that measures and **engineers** the geometry of time across scale. Inside stable bins, the

characteristic time T scales with a size proxy L as $T \propto L^{\alpha}$; the **coherence exponent** α is invariant to clocks and thus comparable across devices, stacks, and labs. With **collapse** as a falsifiable gate and **errors-in-variables** estimation, α becomes a reliable operational signal. Fusing clean, layer-wise slopes yields a real-time $\textbf{ECI}_{\text{QC}}\textbf{(t)}$ that supports **early warnings** (decoherence events) and **design decisions** (scheduler, QEC cadence, modular sizing, I/O offsets).

What this adds. RTM-QC complements fidelity/QEC by introducing a third axis—**tempo geometry**—that explains and controls tails, queues, and synchronization cascades. Modest, reversible controls (batching, staggered resets, micro-jitter, low-variance routing) can **raise** α without degrading throughput or fidelity when used with guardrails.

What it does not do. RTM-QC does not replace physical improvements (fidelities, T_1/T_2), nor does it guarantee causality without the A/B protocols and guardrails we specify. Failures to collapse, high heterogeneity, or regime seams are **informative**, delineating scope boundaries rather than inviting post-hoc fixes.

Near-term agenda.

1. **Run the protocols** (Sec. 8) on superconducting and ion platforms; publish both successes and negatives with full collapse/fusion diagnostics.

2. **Close the loop**: deploy $\text{ECI}_{\text{QC}}(t)$ dashboards and alert playbooks in production; evaluate lead–lag vs. KPI spikes.

3. **Co-design with decoders** and compilers so cadence and routing optimize α subject to throughput/fidelity floors.

4. **Standardize reporting**: figures/tables in Sec. 9, methods YAML, and open robustness artifacts.

Longer-term questions. Incorporate α into **time-changed diffusion models** of queues; develop **holonomy tests** to distinguish curvature from seams; extend to **modular networks** and **neutral-atom** platforms; integrate learning-based controllers that respect collapse gates.

Bottom line. RTM-QC gives quantum teams a **unit-robust, falsifiable lever** over multiscale timing. Measure the slope, **validate by collapse**, fuse when families agree, and **engineer** α—not as a slogan, but as a reproducible practice to deliver more stable and efficient quantum computation.

Appendices

Appendix A — Mathematical Background (RTM essentials for QC)

A.1 Semigroup → power law

Assume binwise scale semigroup $T(bL) = f(b)T(L)$, $f(1) = 1$, and measurability near $b = 1$. Then $f(b) = b^\alpha$ and

$$T(L) = \kappa L^\alpha, v(u) = \log T = \alpha u + \log \kappa, u = \log L.$$

α is **gauge-invariant**; κ is a **clock**.

A.2 1-form & collapse

Define the RTM 1-form $\omega = dv - \alpha\, du$. **Collapse** (residual independence of $v - \alpha u$ from u) is equivalent to **exactness** of ω on a simply connected bin:

$$\omega = d\psi(x), d\omega = 0, \psi \text{ independent of } u.$$

If $\alpha = \alpha(x, u)$, then $d\omega = -d\alpha \wedge du$; nonzero curvature breaks collapse.

A.3 Variable exponents (finite-window bias)

For slowly varying $\alpha(u)$:

$$v(u) = \int_{u_0}^{u} \alpha(s)\, ds + \log \kappa(u), \hat{\alpha}(u; h) = \alpha(u) + O(\varepsilon h),$$

and $R^2_{\text{collapse}} = O((\varepsilon h)^2)$ for window width h.

Appendix B — Estimators & Algorithms

B.1 Orthogonal Distance Regression (TLS/ODR)

Minimize orthogonal residuals:

$$\min_{\alpha,c} \sum_i \frac{(y_i - \alpha x_i - c)^2}{\sigma_{y,i}^2 + \alpha^2 \sigma_{x,i}^2}.$$

Init: Theil–Sen; **CIs:** bootstrap pairs/cluster; **checks:** condition number $< 10^4$; max leverage $< 25\%$.

B.2 Theil–Sen

Median of pairwise slopes $\alpha_{ij} = (y_j - y_i)/(x_j - x_i)$; robust to outliers; mild EIV attenuation.

B.3 SIMEX (optional)

If $\sigma_\xi^2 = \text{Var}(\xi)$ is estimable, simulate $x^{(\lambda)} = x^{obs} + \sqrt{\lambda}\tilde{\xi}$ and extrapolate $\hat{\alpha}(\lambda)$ to $\lambda = -1$.

B.4 Collapse gate

Regress residuals $\tilde{y} = y - \hat{\alpha}x - \hat{c}$ on x; require $R^2_{\text{collapse}} < 0.05$ and flat LOESS; pass clock placebo.

Appendix C — Protocol Cards (copy–paste templates)

C.1 Physical (staggered resets + readout waves)

- **L/T:** L =active cluster size; T = stable calibration interval.
- **Arms:** Control vs RTM-aware (waves + 2–4% reset offsets).
- **Duration:** 2–4 weeks, interleaved.
- **Success:** $\Delta\alpha_{\text{phys}} \geq 0.15$ (95% CI excludes 0), throughput loss ≤5%, collapse passes.

C.2 QEC (micro-jitter / multi-phase)

- **L/T:** $L = d$; T = cycles to logical failure.
- **Arms:** Fixed period vs $Pk = P(1 + \eta k)$, $| \eta k | \leq 0.02$ or 2–3 phase groups.
- **Success:** $\Delta\alpha_{\text{QEC}} \geq 0.15$, no logical-error regression (>5%) at fixed d.

C.3 Runtime (batching + low-variance routing)

- **L/T:** L = post-mapping width; T = makespan.
- **Arms:** Baseline vs wavefront + variance-penalized routing.
- **Success:** $\Delta\alpha_{\text{runtime}} \geq 0.10$ and p95/p50 latency ↓ ≥10%.

C.4 I/O (phase-offset windows)

- **L/T:** L = multiplexing degree; T = readout latency p95 (and p50).
- **Arms:** Synchronous vs phase-offset windows.
- **Success:** $\Delta\alpha_{\text{IO}} \geq 0.10$, p95/p50 ≤ 1.6 over majority of L.

Appendix D — Methods YAML (skeleton)

```yaml
bin:
  platform: "SC"          # or "IONS", "NA"
  temperature_band: "10-15mK"
  firmware_hash: "fpga_1.4.2_dsp_0.9.8"
  topology_id: "mesh-v3"
  routing_policy: "baseline"  # or "rtm-aware"
  syndrome_cadence: "P=3.2us, jitter=0%"
  utilization_band: "30-60%"

estimation:
  min_L_points: 6
  min_logL_span: 0.6
  eiv: "odr"
  odr:
    init: "theil-sen"
    leverage_cap: 0.25
    bootstrap: {clusters: true, reps: 2000, seed: 123}
  simex:
    enabled: false
    lambda: [0.5,1.0,1.5,2.0]
collapse:
  r2_threshold: 0.05
  placebo_clock: true
  changepoint_scan: {method: "PELT", penalty: "BIC"}

fusion:
  heterogeneity_gate_I2: 0.5
  tau2_method: "REML"
  min_families: 2

eci_rt:
```

window_logL: 0.8

horizon_days: 14

smoothing: "median3"

alert:

 z_advisory: -1.5

 z_watch: -2.0

 z_warning: -2.5

Appendix E — Notation Glossary

- L: scale proxy (layer-specific); $u = \log L$.

- T: characteristic time; $v = \log T$.

- α: **coherence exponent** (slope; clock-invariant).

- **Bin**: environment slice with fixed {platform, temperature band, firmware hash, topology ID, routing policy, syndrome cadence, utilization band}.

- **Collapse**: $R^2(\tilde{y} \sim x) < 0.05$ for $\tilde{y} = y - \hat{\alpha}x$; residuals show no trend vs x.

- $\textbf{ECI}_{\text{QC}}(t)$: fused slope via random-effects at time t.

- Q, I^2, τ^2: heterogeneity statistics for fusion.

- ODR/TLS, Theil–Sen, SIMEX: slope estimators under EIV.

- **Adiabatic window**: width h in u where $| \partial_u \alpha | h \ll 1$.

Appendix F — Reproducible Figure Recipes (minimal)

- **Collapse panel**:

  - Fit ODR; compute residuals $\tilde{y}$.

  - Plot y vs x + ODR band; residual vs x with LOESS.

  - Annotate R^2_{collapse}, #L, span, leverage.

- **Forest plot**:

  - For accepted families, display $\hat{\alpha}_f \pm$ CI; compute $w_f, Q, I^2, \hat{\tau}^2$.

- o   Overlay fused $\hat{\alpha}_{QC}$.
- **ECI$_{QC}(t)$:**
    - o   Rolling fusion; show 50/95% bands; background colored by I^2 tiers; mark alert tiers.

APPENDIX G — Empirical Analysis: Quantum Hardware Scaling and the Generational Confounder

G.1. Methodology

To rigorously evaluate the RTM scaling equation ($T \propto L^{\alpha}$) within quantum hardware, we analyzed the coherence times (T_2) of 31 superconducting quantum processors developed by IBM between 2017 and 2026.

- **Sample:** 31 processors ranging from 5 qubits (Canary family) to 1121 qubits (Condor), spanning 5 orders of magnitude in system complexity.

- **Metric:** The spatial scale (L) was defined by the number of physical qubits (N), while the characteristic time was defined by the average processor dephasing time (T_2).

- **Control Variable:** Processors were segmented into strictly defined technological eras (2017-2019, 2020-2022, 2023-2026) to mathematically isolate true topological scaling from materials and fabrication improvements.

G.2. Results: The Confounding Illusion and True Scaling

The analysis exposes a critical statistical confounder in standard quantum hardware evaluations:

1. **The Raw Illusion (Confounded):** A direct regression across the entire 31-processor dataset yields a seemingly positive exponent of $\boldsymbol{\alpha = +0.23 \pm 0.08}$ ($R^2 = 0.20$). This incorrectly implies that larger quantum chips naturally sustain coherence longer.

2. **True Topological Scaling (Same-Generation):** When RTM isolates the data by technology generation—holding the manufacturing quality constant—the true physical scaling emerges. The exponent undergoes a fundamental inversion, yielding $\boldsymbol{\alpha = -0.24}$ for the 2020-2022 era, and $\boldsymbol{\alpha = -0.39}$ for the 2023-2026 era.

G.3. Topological Classification: The Inverse Regime

By converging on a true exponent of $\alpha \approx -0.35$, RTM classifies macroscopic quantum decoherence into the **Inverse Transport Class** ($\alpha < 0$). This places quantum hardware scaling in the exact same mathematical category as Stokes-Einstein molecular diffusion ($\alpha = -1.19$). Physically, this proves that decoherence in large processor lattices is a **collective** phenomenon. As the topological area of the system expands, it unavoidably accumulates crosstalk, two-level system (TLS) defects, and correlated environmental noise. The sheer size of the system actively works against its ability to maintain a coherent state.

G.4. Conclusion

The expanded empirical analysis of 31 IBM processors successfully validates the RTM framework's diagnostic power. It proves that the apparent coherence improvements in modern monolithic quantum chips are solely the result of heroic engineering overcoming a fundamentally hostile physical scaling law ($\alpha < 0$). This RTM signature perfectly justifies the quantum computing industry's recent strategic pivot away from infinitely larger monolithic chips, and toward modular, multi-chip architectures designed to artificially bypass this negative topological scaling limit.