

Memory Physics, Finite Capacity, and Topology in Heterogeneous Quantum Networks

David Vesterlund

david@vesterlundventures.se

Vesterlund Ventures Holding AB <https://orcid.org/0009-0000-6455-1141>

Research Article

Keywords: quantum networks, quantum memory, discrete-event simulation, finite capacity, heterogeneous placement, reproducibility, memory topology, quantum information

Posted Date: October 13th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11330914/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.
[Read Full License](#)

Additional Declarations: The authors declare no competing interests.

Memory Physics, Finite Capacity, and Topology in Heterogeneous Quantum Networks

David Vesterlund¹ 

¹ Vesterlund Ventures, WestQuant Open; david@vesterlundventures.se

* Correspondence: david@vesterlundventures.se

Abstract

Quantum-network capacity planning couples memory physics, finite storage, topology, and traffic. We study these interactions using a discrete-event simulator with six representative memory profiles. A memory–topology atlas contains 9,360 completed runs, complemented by 8,730 confirmatory runs and an extended per-node memory-budget sweep. We distinguish a threshold relative to the best performance observed on a finite grid from a physical saturation limit. Among 840 reported threshold estimates, 22.3% reach 95% of the observed maximum by four slots per node, while 35.5% first reach it at the upper grid boundary of 64 slots. Four tested scalar reductions do not provide a topology-independent collapse; the best pooled quadratic fit explains 37.9% of response variance. The retrieval-window parameter $\chi_r = \tau_r p_{\text{gen}}/t_{\text{attempt}}$ provides an interpretable descriptor, but an independent-channel approximation does not establish a network memory-sizing law. In 5,310 heterogeneous-placement runs, hub and betweenness-based placement outperform peripheral placement in aggregate, with topology-specific exceptions. Coupling assumptions change absolute performance while preserving the aggregate strategy ordering. Harmonized model comparisons provide limited robustness evidence, not independent native-protocol validation. The results support model-specific capacity and placement analysis with explicit metric definitions, finite-grid limitations, and interface assumptions.

Keywords: quantum networks; quantum memory; discrete-event simulation; finite capacity; heterogeneous placement; reproducibility

1. Introduction

Quantum networks promise distributed quantum computation, secure communication, and distributed sensing [1]. A central engineering question is not simply *which memory technology is best*, but rather: *how do physical memory properties, finite capacity, network position, topology, and offered load jointly determine useful end-to-end entanglement?* This question couples device-level physics (coherence time, retrieval efficiency, readout fidelity, physical interfaces) to network-level structure (topology, routing, contention), and its answer determines hardware–network co-design rules for heterogeneous quantum networks. Figure 1 summarizes the key findings of this study.

Existing analytical results characterize scaling and rate–memory trade-offs for specific repeater architectures, often homogeneous chains with prescribed memory models [2,3,5]. Simulation frameworks such as SeQUeNCe [6], NetSquid [7], and QuISP [8] enable more detailed studies, while recent cross-validation work shows that simulator assumptions can materially affect timing even when fidelity agrees [9]. Recent work on

Received:

Revised: 8 October 2026

Accepted:

Published:

Copyright: © 2026 by the authors.

Submitted to *Quantum Rep.* for possible open access publication under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

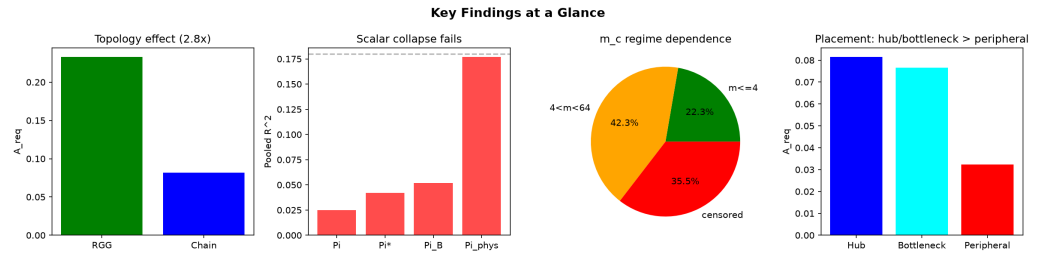


Figure 1. Key findings at a glance. The retrieval-window parameter $\chi_r = \tau_r p_{\text{gen}}/t_{\text{attempt}}$ (where τ_r is the retrieval decay time, not T_2) provides an interpretable descriptor but does not establish a universal memory-sizing law; the 95%-of-observed-maximum memory threshold is strongly regime-dependent with 35.5% of regimes as boundary cases at $m = 64$; and hub/bottleneck placement outperforms peripheral placement in aggregate under the tested heterogeneous configurations.

heterogeneous link architectures [10] and hybrid repeater designs [11,12] motivates the question of whether conclusions derived from homogeneous models transfer to heterogeneous hardware.

Simulation-based network-design conclusions are valuable only if they are not artifacts of a single implementation. We therefore supplement the component-level analytical validation with a harmonized cross-implementation robustness check (Sec. 4.3), in which selected conclusions are re-evaluated through harmonized implementations executed on the SeQUeNCe and SimQN event frameworks. The purpose is to test whether selected structural conclusions survive an independent implementation/event-framework pathway, not to establish universal simulator invariance. A companion reliability study reports the full simulator-invariance methodology.

This paper addresses four research questions:

RQ-A1 Which memory- and network-level parameters limit end-to-end entanglement delivery?

RQ-A2 How does performance vary across representative memory profiles and topologies?

RQ-A3 Does a universal scaling law for memory sufficiency exist?

RQ-A4 How does heterogeneous memory hardware affect conclusions?

Our central thesis is:

Within the investigated operating regimes, the per-node memory-budget threshold m_c is strongly regime-dependent and does not collapse to a topology-independent scalar. The retrieval-window parameter $\chi_r = \tau_r p_{\text{gen}}/t_{\text{attempt}}$ (where τ_r is the retrieval decay time, not the coherence time T_2) provides an interpretable physical descriptor, but an independent-channel approximation does not establish a network memory-sizing law. Conclusions about heterogeneous memory deployment additionally require explicit interface modeling.

We do **not** claim that no universal scaling law can exist, that one memory technology is universally superior, or that the observed saturation behavior constitutes a fundamental physical law. Our claims are restricted to the tested models, parameter ranges, and operating regimes.

The retrieval-window descriptor. A physical timescale descriptor used in this paper is the coincidence-window parameter

$$\chi_r = \frac{\tau_r p_{\text{gen}}}{t_{\text{attempt}}}, \quad (1)$$

where τ_r is the retrieval decay time (not the coherence time T_2), p_{gen} is the per-attempt link generation success probability, and t_{attempt} is the link attempt interval. The parameter χ_r counts the expected number of successful link generations within one memory lifetime and separates two operating regimes:

Coincidence-limited ($\chi_r \ll 1$): Link success within the retrieval lifetime is rare. The single-link memory threshold is

$$m \approx \frac{\ln(1 - \alpha)}{\ln(1 - p_w)}, \quad p_w = 1 - (1 - p_{\text{gen}})^{\tau_r / t_{\text{attempt}}}, \quad (2)$$

where p_w is the probability of at least one link success during the memory lifetime and α is the target fraction of peak performance.

Queue-limited ($\chi_r \gg 1$): Many link successes occur within the retrieval lifetime, and memory demand is governed by Little's law ($m \propto \text{load} \times \text{wait time}$) rather than by coincidence probability.

For multi-hop paths of length h , the path-success probability is $p_{\text{path}} = p_w^h$ (all links must succeed), so the memory threshold scales with hop length. Table 1 maps the four primary memory profiles to their χ_r values and regimes.

Table 1. Regime map for the four primary memory profiles at atlas parameters ($p_{\text{gen}} = 0.005$, $t_{\text{attempt}} = 10^{-3}$ s). $\chi_r = \tau_r \cdot p_{\text{gen}} / t_{\text{attempt}}$.

Profile	χ_r	Regime
Trapped Ion	50	Queue-limited ($\chi_r \gg 1$)
Neutral Atom	0.5	Transitional ($\chi_r \sim 1$)
Diamond (Electron)	0.22	Coincidence-limited ($\chi_r \ll 1$)
AFC	0.025	Coincidence-limited ($\chi_r \ll 1$)

The retrieval-window descriptor χ_r provides a physical interpretation of the regime-dependent m_c distribution (Sec. 8). No predictive accuracy or invariance rate is asserted for χ_r in this revision; the independent-channel approximation is an illustrative reference, not a validated network memory-sizing law.

2. Related Work

2.1. Quantum Repeater Scaling

Scaling behavior for quantum repeaters has been analyzed under a range of homogeneous-chain assumptions. Razavi, Piani, and Lütkenhaus [2] showed that finite memory coherence can change favorable polynomial distance scaling into an exponential degradation in $\sqrt{L}$ in their optimized repeater model. More recent work studies exact rate analysis with imperfect memories [4] and joint optimization of repeater count, spacing, and distillation under rate and fidelity constraints [5]. These results provide important benchmarks, but they do not directly address the cross-product of heterogeneous memory profiles and network topology studied here.

2.2. Topology and Entanglement Distribution

The impact of network topology on entanglement distribution has been studied by several groups. Ooi, Sutcliffe, and Beghelli [13] identified four topology-dependent performance regimes across 81 real topologies. Ozaydin et al. [14] developed optimal algorithms for multi-ring topologies. Vardoyan et al. [15] derived end-to-end capacity bounds via flow maximization. Pirandola [19] derived end-to-end capacity bounds for quantum networks. Takeoka et al. [20] established the fundamental rate-loss tradeoff for optical

quantum communication, constraining the achievable rates that any network design must respect. Vasantam and Towsley [21] analyzed the stability of quantum networks with max-weight scheduling, providing a queueing-theoretic foundation for resource contention analysis. These studies focus on routing, capacity, and scheduling rather than memory-physics interactions.

2.3. Heterogeneous Quantum Networks

Recent work has begun exploring heterogeneous hardware. Soon et al. [10] demonstrated that mixed link architectures work without significant rate loss using QuISP. Van Loock et al. [11] and Liu et al. [12] proposed hybrid repeater architectures combining trapped ions, neutral atoms, and AFC memories. These studies motivate, but do not replace, a controlled comparison of representative memory profiles across multiple topology classes and placement rules.

2.4. Resource Placement

Repeater and memory placement has been formulated as an optimization problem. Pouryousef et al. [16] developed a network-planning framework that allocates repeaters and memories while accounting for rate, fidelity, multiplexing, and coherence. Sripotchanart et al. [17] proposed a scalable two-step linear-programming approach for repeater placement, while Islam and Arslan [18] developed fast heuristic deployment models. Our focus differs: we hold the network-planning objective fixed and study how heterogeneous memory quality, finite capacity, topology, and interface assumptions interact.

2.5. Simulator Validation

Cross-validation of quantum network simulators is an emerging concern. Chung et al. [9] cross-validated QuISP and SeQUeNCe, finding that different simulator assumptions can produce different timing results even when fidelity results agree after harmonization. This motivates careful validation of any single-simulator study. The present work supplements its component-level analytical validation with a harmonized cross-implementation robustness study (Sec. 4.3). We emphasize that this cross-implementation check does not yet use native SeQUeNCe or SimQN protocols, does not include NetSquid or QuISP, and that the SeQUeNCe and SimQN adapters share a common harmonized model. The purpose is therefore to test whether selected conclusions survive an independent implementation/event-framework pathway, not to establish universal simulator invariance.

3. Quantum Memory and Network Model

3.1. Representative Memory Profiles

We model six representative quantum-memory profiles using literature-informed parameter bundles (Table 2):

These profiles are *representative bundles*, not complete device-level hardware comparisons. Some bundles combine characteristic values reported for related implementations within a memory platform; Supplemental Table S1 therefore labels each numerical entry as MEASURED, DERIVED, or ASSUMED. The simulated budget m is an abstract number of independently usable memory slots per node and is not identified with an experimentally reported multimode capacity.

Table 2. Representative memory-profile parameters. T_1 and T_2 are in seconds, η_r is retrieval efficiency at $t = 0$, τ_r is the retrieval decay time (s), and F_{ro} is readout fidelity. Provenance and source citations are provided in Supplemental Table S1 [22–33].

Profile	T_1 (s)	T_2 (s)	η_r	τ_r (s)	F_{ro}
Trapped Ion	100	10.0	0.90	10.0	0.999
Neutral Atom	1.0	0.1	0.80	0.5	0.99
Diamond (Electron)	1.0	0.044	0.70	0.1	0.95
Diamond (Nuclear)	100	1.3	0.50	1.0	0.95
AFC	10	0.005	0.50	0.01	0.99
Superconducting	0.001	0.0001	0.95	0.001	0.999

3.2. Network Topologies

We study nine topology classes: chain, star, random geometric graph (RGG), Waxman, hierarchical, lattice, scale-free, binary tree, and ring. Topologies are generated with controlled parameters (node count, edge density, geographic extent).

3.3. Simulator

Our discrete-event simulator models:

- Per-node memory capacity via a *placement* dictionary
- Homogeneous or heterogeneous memory profiles
- Homogeneous or heterogeneous decoherence models
- Depolarizing, dephasing, or amplitude-damping noise
- Serial or parallel link generation
- Entanglement swapping with configurable success probability
- Retrieval efficiency and readout fidelity
- Deadline-based service qualification
- Poisson request arrivals with optional traffic matrices
- Warmup and censoring to reduce transient artifacts

Table 3 lists the default simulator parameters.

Table 3. Simulator parameters used throughout the paper unless otherwise stated.

Parameter	Value	Description
p_{gen}	0.005	Link generation success probability
t_{attempt}	10^{-3} s	Link attempt interval
Deadline	1.0 s	Request deadline
F_{min}	0.5	Minimum fidelity for qualification
p_{swap}	0.98	Entanglement swap success probability
F_{initial}	0.9	Initial link fidelity
T_{obs}	200 s	Simulation observation time
Warmup	First 10% of requests	Warmup period
n_{requests}	100	Requests per simulation

Noise levels are defined for the sensitivity analysis (Sec. 10) as: **low** = $0.1 \times$ baseline depolarizing rate, **medium** = $1.0 \times$ baseline, **high** = $10 \times$ baseline. The baseline depolarizing rate is derived from the memory profile’s T_2 . In the atlas and confirmatory experiments, noise is not independently varied; the depolarizing rate is fixed by each profile’s T_2 .

Run-length caveat: With $n_{\text{requests}} = 100$ and the highest traffic level $\lambda = 50$, the observation window covers approximately $100/50 = 2$ s of simulated time. This is short

relative to the $T_{\text{obs}} = 200$ s parameter and may not reach full stationarity for queueing dynamics at high load. The warmup period (first 10% of requests) partially mitigates transient effects, but the high-traffic results should be interpreted with the caveat that the system may not have fully reached steady state.

The primary performance metric is A_{req} : the fraction of *effective incoming requests* (excluding warmup, censored, and no-path requests) that are completed within deadline. A companion metric A_{qual} additionally requires the final fidelity to exceed F_{min} . Quality-qualified throughput showed stronger ordinal agreement across the tested harmonized implementations than completion-only throughput. Because request-processing and denominator conventions were not fully identical across implementations, we treat this as a robustness observation rather than as a general superiority claim for either metric. All confidence intervals reported in this paper are 95% bootstrap intervals (10,000 resamples, percentile method) computed across seeds unless otherwise stated.

Definition of m : Throughout this paper, m denotes the *per-node memory budget*: every node is assigned m memory slots. This definition is used consistently in the atlas, the confirmatory replication, and the m_c audit.

3.4. Mathematical Model

The simulator's quantum-state and noise models are defined as follows.

Werner-state parameterization. Each entangled link is represented as a Werner state with fidelity F to the target Bell state $|\Phi^+\rangle$:

$$\rho = F |\Phi^+\rangle\langle\Phi^+| + \frac{1-F}{3} \sum_i |\Psi_i\rangle\langle\Psi_i|, \quad (3)$$

where $\{|\Psi_i\rangle\} = \{|\Phi^-\rangle, |\Psi^+\rangle, |\Psi^-\rangle\}$ are the other three Bell states. The Werner state is entangled iff $F > 0.5$.

Decoherence channels. During memory storage, the depolarizing channel drives F toward the completely mixed value $F = 0.25$:

$$F(t) = 0.25 + (F_0 - 0.25) \exp(-\Gamma t), \quad (4)$$

where Γ is the depolarizing rate derived from the memory profile's T_2 . The simulator also supports pure dephasing and amplitude-damping channels; in the atlas and confirmatory experiments, the depolarizing channel is used with Γ determined by T_2 .

Entanglement-swap composition. Multi-hop entanglement is established via Bell-state-measurement (BSM) swaps with success probability p_{swap} . The simulator composes Werner states via the Werner parameter $w = (4F - 1)/3$. For two links with Werner parameters w_1 and w_2 and a BSM with Werner parameter $w_{\text{bsm}} = (4F_{\text{bsm}} - 1)/3$, the output after a successful swap is

$$w_{\text{new}} = w_1 w_2 w_{\text{bsm}}, \quad F_{\text{swap}} = \frac{3 w_{\text{new}} + 1}{4}, \quad (5)$$

where F_{bsm} is the BSM fidelity (default $F_{\text{bsm}} = 0.99$). For an ideal BSM ($F_{\text{bsm}} = 1$, i.e., $w_{\text{bsm}} = 1$), this reduces to the standard Werner-state swap formula

$$F_{\text{swap}} = F_1 F_2 + \frac{(1 - F_1)(1 - F_2)}{3}. \quad (6)$$

The swap succeeds with probability p_{swap} per hop. Failed swaps consume the stored pairs.

Retrieval efficiency. When a stored pair is retrieved from memory, the retrieval succeeds with probability given by an exponential decay model

$$\eta(t) = \eta_r \exp\left(-\frac{t}{\tau_r}\right), \quad (7)$$

where η_r is the profile-dependent retrieval efficiency at $t = 0$ and τ_r is the retrieval decay time (both listed in Table 2). The simulator also supports a Gaussian model $\eta(t) = \eta_0 \exp[-(t/\tau)^2/2]$ for AFC-type memories, but the exponential model is used throughout the atlas and confirmatory experiments. Readout fidelity F_{ro} is applied as a depolarizing channel: with probability F_{ro} the state is preserved, and with probability $1 - F_{ro}$ the state is fully depolarized to the completely mixed state ($F = 0.25$):

$$F \leftarrow F_{ro} \cdot F + (1 - F_{ro}) \cdot 0.25. \quad (8)$$

T1/T2 compatibility. All memory profiles satisfy the physical constraint $T_2 \leq 2 T_1$, which follows from the relation $1/T_2 = 1/(2T_1) + 1/T_\phi$ with $T_\phi \geq 0$ (the pure dephasing time T_ϕ is non-negative). This is a constraint on the parameter space, not a derivation of a specific value; the T_2 values in Table 2 are drawn from experimental literature and independently satisfy this inequality.

Readout-model note. The frozen atlas, confirmatory, and m_c datasets (commit 1f86a98, October 4) were generated using a *multiplicative* readout model $F_{\text{out}} = F_{ro} \cdot F$. The current codebase has since been updated to a *depolarizing* readout model $F_{\text{out}} = F_{ro} \cdot F + (1 - F_{ro}) \cdot 0.25$, which is the model described in Eq. (8) above. The difference between the two models is 1–2.5% in fidelity (systematic: the depolarizing model yields higher fidelity because it mixes toward the completely mixed state at fidelity 0.25, which is above the multiplicative floor $F_{ro} \cdot F$ when $F > 0.25$). All quantitative results reported in this paper are drawn from the frozen datasets and therefore reflect the multiplicative readout model. This discrepancy does not affect the retrieval-window descriptor χ_r (Sec. 8) because χ_r depends on τ_r and generation parameters ($p_{\text{gen}}, t_{\text{attempt}}$), not on readout fidelity. Regenerating the frozen datasets with the depolarizing model is future work.

4. Simulator Validation

4.1. Component-Level Analytical Validation

We validate the simulator’s component models against analytical formulas for elementary link generation probability, Werner-state decoherence, multi-hop swapping fidelity composition, and retrieval efficiency/readout. The implementation is verified against the formulas (numerical error $< 10^{-10}$).

4.2. Component-Level Reference Comparison

As an additional reference-model check, we evaluate five scenario families (V1: elementary link, V2: three-node repeater, V3: multi-hop, V4: finite memory, V5: decoherence crossover) using parameterizations aligned with SeQUeNCe component models. This exercise is *not* a protocol-level cross-simulator validation because the SeQUeNCe protocol stack is not executed end-to-end. Accordingly, we use these results only as component-level validation and do not infer protocol-level agreement. Table 4 shows the component-level reference comparison.

Table 4 shows that fidelity agreement is moderate (median $|\Delta F| = 0.065$) but ranking correlation varies substantially by scenario, from $\rho = 0.32$ (V2) to near zero (V4, V5). The low ranking correlation in V4 and V5 reflects differences in how the two models handle

Table 4. Component-level reference comparison per scenario. ρ is the Spearman rank correlation between WestQuant and reference-model A_{req} .

Scenario	n	Median $ \Delta F $	Median $ \Delta A $	ρ
V1: Elementary link	270	0.017	0.178	0.218
V2: Three-node repeater	270	0.058	0.356	0.319
V3: Multi-hop	360	0.103	0.378	0.130
V4: Finite memory	270	0.152	0.356	0.027
V5: Decoherence crossover	540	0.042	0.244	0.025
Overall	1,710	0.065	0.152	—

finite-memory and decoherence effects. A matched protocol-level comparison against an independent codebase remains future work.

4.3. Cross-Implementation Robustness Check

A cross-implementation robustness check was performed using harmonized implementations on the SeQUeNCe and SimQN event frameworks. In summary: the topology ordering (RGG > star > chain) and the memory-profile ordering (trapped ion > neutral atom > diamond electron > AFC) were reproduced under A_{qual} , while absolute performance values and finite-memory contention were implementation-sensitive. The SeQUeNCe and SimQN adapters share a common harmonized model and do not constitute two independent physical-model replications; native SeQUeNCe/SimQN protocols were not included. This check therefore provides limited robustness evidence, not independent native-protocol validation.

5. Representative Memory-Profile Atlas

5.1. Experimental Design

The atlas spans 6 representative memory profiles, 9 topology classes, 3 traffic levels ($\lambda \in \{1, 10, 50\}$), 3 network sizes ($N \in \{10, 20, 50\}$), 4 memory budgets ($m \in \{1, 2, 4, 8\}$, per node), and 5 random seeds. Noise is not independently varied in the atlas; the depolarizing rate is determined by each memory profile's T_2 (Sec. 3). The full Cartesian design contains $6 \times 9 \times 3 \times 3 \times 4 \times 5 = 9,720$ nominal cells, of which 9,360 completed runs are retained. The 360 missing cells (3.7%) arise from disconnected random-geometric graphs at small N and from simulation failures; no missing cells are imputed. The frozen atlas dataset is `v05_fixes/results/p3_scalar_test.csv`.

5.2. Confirmatory Replication

A pre-specified confirmatory campaign spanning 6 profiles, 5 topologies (chain, star, RGG, lattice, scale-free), 3 traffic levels, 5 budgets $m \in \{1, 2, 4, 8, 16\}$, and 20 seeds produced a frozen dataset of 8,730 completed simulations (frozen file: `v05_fixes/results/p2_c3_extended.csv`). The full Cartesian design contains $6 \times 5 \times 3 \times 5 \times 20 = 9,000$ cells; 270 cells (3.0%) are missing due to disconnected random-geometric graphs and simulation failures. No missing cells are imputed. The completed dataset confirms:

- **C1 (family effect):** Memory family materially affects A_{req} (Trapped Ion mean= 0.453 vs Superconducting mean= 0.003). The tested cross-implementation analysis reproduced the ordering of the evaluated memory profiles under the fidelity-qualified metric, whereas absolute performance differences were implementation-sensitive.

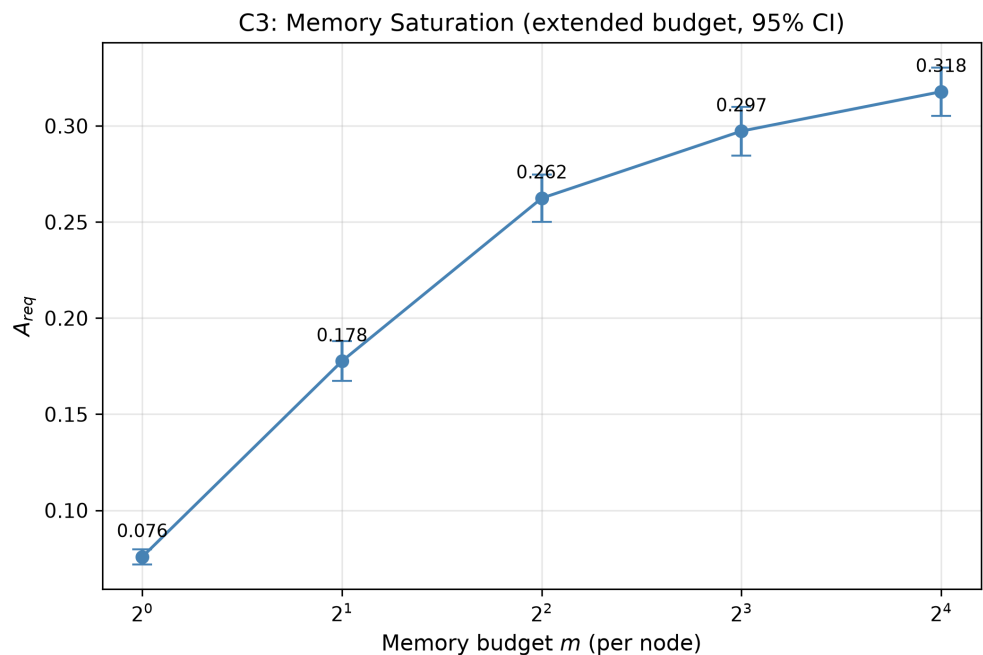


Figure 2. C3 memory-budget response with extended budget ($m = 1, 2, 4, 8, 16$, per node). Error bars show 95% CI across 20 seeds. The curve flattens but does not reach a clear plateau within the tested range.

- **C2 (topology effect):** Topology materially affects A_{req} (RGG mean= 0.233 vs chain mean= 0.082). This comparison is confounded with path length and total memory: under the per-node budget definition, topologies with more nodes receive more total memory, and RGGs have shorter mean path lengths than chains. The RGG > star > chain ordering was reproduced in the harmonized cross-implementation experiment.
- **C3 (diminishing returns):** Memory-budget response shows diminishing marginal gains but the 95%-of-observed-maximum threshold is not fully resolved at $m = 16$ ($m = 1$: 0.076, $m = 2$: 0.178, $m = 4$: 0.262, $m = 8$: 0.297, $m = 16$: 0.318). Monotonicity of $A(m)$ was reproduced in the cross-implementation check; the diminishing-returns conclusion rests on the finer WestQuant sweep and was not resolved by the coarser cross-implementation grid.

Figure 2 shows the extended budget-response curve. The curve is nondecreasing and the marginal gain diminishes, but a clear asymptotic plateau is not reached within $m \leq 16$.

Data provenance: All quantitative results in this paper are drawn from the frozen v05_fixes dataset directory, not from the earlier exploratory experiments in the a07 or a10 directories. The atlas uses p3_scalar_test.csv (9,360 runs), the confirmatory replication uses p2_c3_extended.csv (8,730 runs), and the m_c audit uses p1_mc_values.csv (840 m_c estimates per α). Earlier exploratory runs (a07: 10,080 raw results, 2,817 m_c estimates; a10: 5,238 confirmatory runs) used different parameter grids and are not the basis for the numbers reported here.

6. Memory-Budget Response and Memory-Budget Threshold

6.1. Budget-Response Curves

Figure 3 shows memory-budget response curves for four memory families across three topologies, using the per-node budget definition and an extended sweep $m \in \{1, 2, 4, 8, 16, 32, 64\}$.

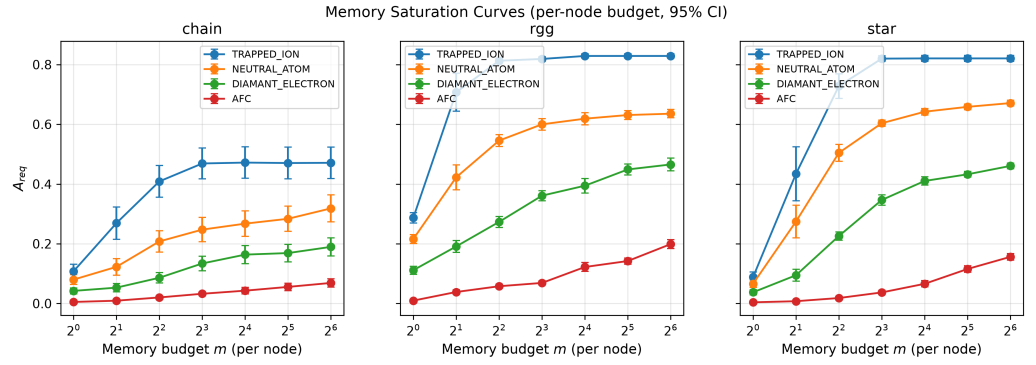


Figure 3. Memory saturation curves with 95% CI for four memory families across three topologies (per-node budget $m \in \{1, 2, 4, 8, 16, 32, 64\}$). Trapped-ion profiles approach a plateau within the tested range; AFC profiles are still increasing at $m = 64$.

6.2. Memory-Budget Threshold (m_c)

We define the memory-budget threshold as the smallest per-node budget at which the primary performance metric A_{req} reaches a specified fraction of the best observed performance on the tested budget grid:

$$m_c(\alpha) = \min \left\{ m : A_{\text{req}}(m) \geq \alpha \cdot A_{\text{max}}^{\text{obs}} \right\} \quad (9)$$

with primary $\alpha = 0.95$. Here $A_{\text{max}}^{\text{obs}} = \max_m A_{\text{req}}(m)$ is the maximum over the *observed* budget grid $m \in \{1, 2, 4, 8, 16, 32, 64\}$, not an asymptotic limit. Because $A_{\text{max}}^{\text{obs}} > 0$ whenever any budget produces a nonzero signal, some budget on the grid always satisfies the criterion; the threshold is therefore always finite on the grid. We exclude regimes where $A_{\text{max}}^{\text{obs}} < 0.02$ (no meaningful signal), which removes 60 of the 900 profile–topology–traffic–size–seed combinations before the budget dimension. (The companion metric A_{qual} was used in the cross-implementation check because it showed stronger ordinal agreement across implementations, but the primary m_c definition uses A_{req} , consistent with the primary metric declared in Sec. 3.)

Three distinct concepts should not be conflated:

1. **Grid threshold:** The first budget on the discrete grid $\{1, 2, 4, 8, 16, 32, 64\}$ that reaches $\alpha \cdot A_{\text{max}}^{\text{obs}}$. This is what m_c measures. A threshold at $m_c = 16$ does not imply that the minimum integer budget is exactly 16; the true minimum lies in $[8, 16]$ given the grid resolution.
2. **Practical plateau:** Whether the budget-response curve has flattened enough that further budget increases yield negligible improvement. This requires a separate plateau criterion (e.g., $|A(2m) - A(m)| < \delta$) and is not automatically established by m_c .
3. **Asymptotic threshold:** The threshold relative to an asymptotic performance level $A_\infty = \lim_{m \rightarrow \infty} A(m)$, if this limit can be identified. Our data do not identify a stable asymptote for all families; we therefore report grid thresholds, not asymptotic thresholds.

Values reaching the upper boundary of the sweep ($m_c = 64$) indicate that the threshold was not reached at a lower budget on the grid. We label these as *boundary cases* rather than “right-censored,” because the threshold relative to $A_{\text{max}}^{\text{obs}}$ is always defined on the grid; what is unknown is whether the curve would continue to rise beyond $m = 64$.

An audit across 4 families $\times$ 5 topologies $\times$ 3 traffic levels $\times$ 3 network sizes $\times$ 7 budgets $\times$ 5 seeds (6,300 nominal runs, of which 5,880 completed; 420 runs missing due to disconnected RGG graphs; 840 m_c estimates at $\alpha = 0.95$ after excluding 60 low-signal regimes; frozen file: v05_fixes/results/p1_mc_values.csv) shows:

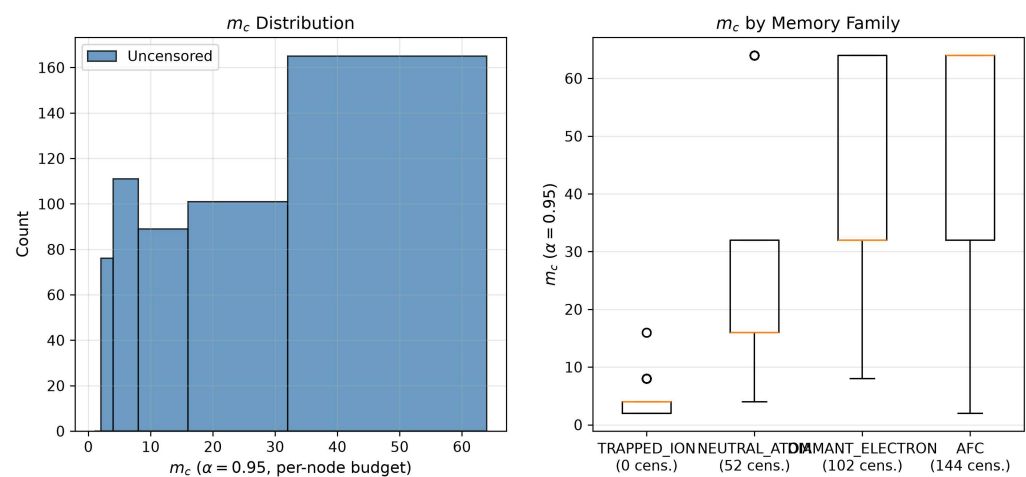


Figure 4. (Left) m_c distribution at $\alpha = 0.95$ (per-node budget). Boundary cases ($m_c = 64$) are shown in red. (Right) m_c by memory family; boundary counts are annotated.

- $m_c = 1$: 0 regimes (0%) 346
- $m_c \leq 4$ (interior): 187 regimes (22.3%) 347
- $4 < m_c < 64$ (interior): 355 regimes (42.3%) 348
- Boundary case ($m_c = 64$): 298 regimes (35.5%) 349

Table 5 shows the per-family m_c distribution, and Figure 4 visualizes the overall distribution. 350

Table 5. Per-family m_c at $\alpha = 0.95$ (per-node budget). “Boundary” indicates $m_c = 64$. 351

Family	Median m_c	Range	Boundary	n
Trapped Ion	4	[2, 16]	0/210	210
Neutral Atom	16	[4, 64]	52/210	210
Diamond (Electron)	32	[8, 64]	102/210	210
AFC	≥ 64	[2, ≥ 64]	144/210	210

The threshold location varies strongly by memory profile: the Trapped Ion profile reaches the 95%-of-observed-maximum threshold at a median $m_c = 4$, while AFC reaches the boundary ($m_c = 64$) in 69% of its regimes. Because 35.5% of observations are boundary cases, we do not report a single mean or standard deviation for m_c . The data support a regime-dependent threshold, not a universal constant. These m_c estimates are model-specific quantitative results of the WestQuant simulation; a coarser cross-implementation sweep reproduced the qualitative regime-dependence of the memory threshold but showed sensitivity of threshold location to implementation details. Direct quantitative cross-validation of the full $m = \{1, 2, 4, 8, 16, 32, 64\}$ threshold analysis remains future work. 352

α stability: The median m_c increases from 16 ($\alpha = 0.90$) to 32 ($\alpha = 0.95$ and 0.99), indicating that the threshold is moderately sensitive to the saturation criterion. 353

7. Failure of Universal Scalar Scaling 364

We test four dimensionless scaling hypotheses. The first three are motivated by dimensional analysis of the load–capacity ratio and its topology corrections; the fourth incorporates memory physics: 365

$$\Pi = \frac{\lambda t_{\text{attempt}}}{m p_{\text{gen}}} \quad (10)$$

$$\Pi^* = \Pi \cdot \sqrt{\bar{h}} \quad (11)$$

$$\Pi_B = \max_v \rho_v, \quad \rho_v = \frac{\Lambda_v}{m_v \cdot \mu}, \quad \mu = \frac{p_{\text{gen}}}{t_{\text{attempt}}} \quad (12)$$

$$\Pi_{\text{phys}} = \frac{\Pi}{\eta_r \cdot \exp(-t_{\text{storage}}/T_2)} \quad (13)$$

where $\bar{h}$ is the mean shortest-path length, Λ_v is the traffic-weighted betweenness of node v , m_v is the memory at node v , and t_{storage} is the mean storage time. Π_B is the maximum node utilization (bottleneck-flow parameter). Π_{phys} incorporates retrieval efficiency η_r and coherence decay.

Quantitative test: We compute the coefficient of determination R^2 for A_{req} as a function of each scalar (log-transformed). All four candidates are evaluated in the pooled data; Π , Π_B , and Π_{phys} are also evaluated separately within each memory profile (Table 6):

Table 6. R^2 of A_{req} versus the three candidates evaluated per profile. Π^* was evaluated in the pooled global collapse test ($R^2 = 0.042$) but was not separately refit per profile in the frozen closure analysis.

Profile	$R^2(\Pi)$	$R^2(\Pi_B)$	$R^2(\Pi_{\text{phys}})$
Trapped Ion	0.153	0.255	0.245
Neutral Atom	0.081	0.176	0.166
Diamond (Electron)	0.040	0.123	0.112
Diamond (Nuclear)	0.024	0.103	0.085
AFC	0.017	0.084	0.067
Superconducting	0.007	0.042	0.000
Pooled R^2	0.025	0.052	0.177

Among the three candidates evaluated per profile, no scalar achieves $R^2 > 0.26$; all four candidates have pooled $R^2 < 0.18$. For Π alone, 85–99% of within-profile response variance remains unexplained, and Figure 5 shows that the residual structure is visibly organized into topology-specific bands. Table 7 quantifies the unexplained fraction per profile.

Table 7. Fraction of within-profile response variance unexplained by the log- Π regression, computed as $1 - R^2(\Pi)$. This quantity is residual variance, not a direct variance-component estimate for topology.

Profile	Unexplained fraction $1 - R^2(\Pi)$
Trapped Ion	0.847
Neutral Atom	0.919
Diamond (Electron)	0.960
Diamond (Nuclear)	0.976
AFC	0.983
Superconducting	0.994

Figure 5 shows the scalar collapse test, and Figure 6 compares the R^2 values across families.

Claim: In the WestQuant dataset, none of the four tested scalar reductions achieves a topology- and hardware-independent collapse. We do not claim that no universal scaling

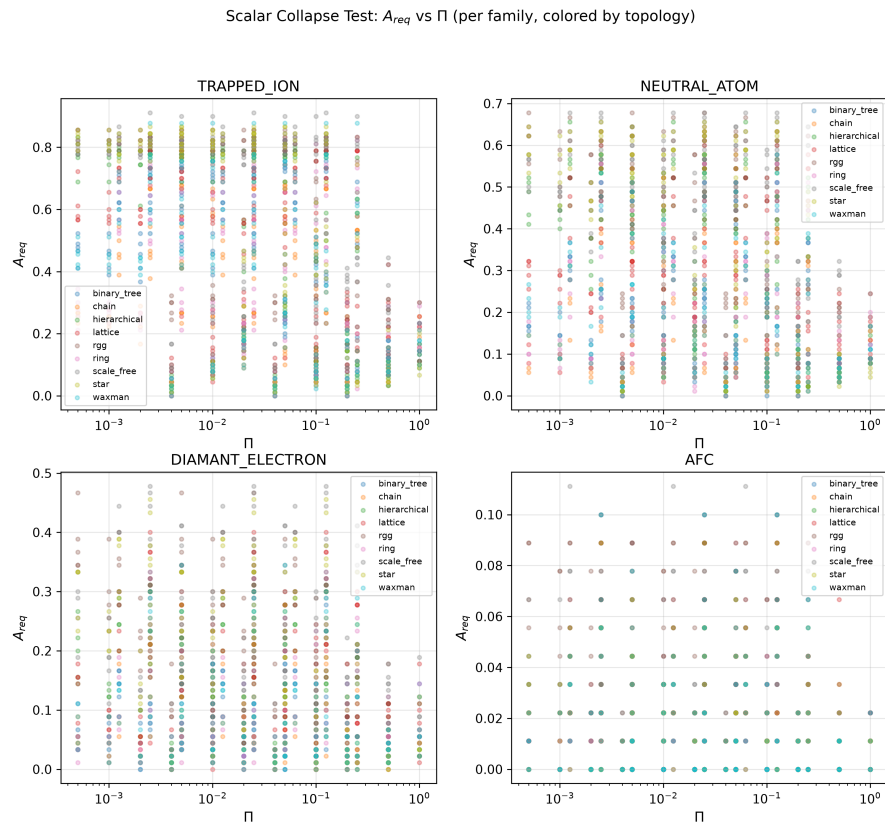


Figure 5. Scalar collapse test: A_{req} vs Π for four memory families, colored by topology. Topology-specific bands persist within each family, confirming that Π does not capture topology-dependent variation even when hardware is held fixed.

law can ever exist—only that these specific candidates are insufficient. The physics-aware candidate Π_{phys} achieves the highest overall R^2 (0.177) but still leaves more than 82% of variance unexplained. The scalar-collapse result has not yet been independently cross-

Nonlinear collapse test: Because the linear R^2 test may miss a nonlinear monotonic relationship, we additionally compute (i) Spearman rank correlation ρ_s between each scalar and A_{req} , and (ii) R^2 of a quadratic log-transformed fit $A_{\text{req}} = a(\log \Pi)^2 + b \log \Pi + c$. Table 8 reports the pooled results. For Π , Π^* , and Π_B , the quadratic R^2 remains below 0.06 and $|\rho_s| < 0.22$, confirming that both linear and nonlinear monotonic fits fail. For Π_{phys} , the quadratic fit improves to $R^2 = 0.379$ (pooled) with $\rho_s = -0.721$, but per-family quadratic R^2 remains below 0.28 (Trapped Ion: 0.274; AFC: 0.069), indicating that the pooled improvement is driven by inter-family separation rather than within-family collapse. We therefore conclude that the failure of universal scalar scaling is robust to nonlinear monotonic transformations for Π , Π^* , and Π_B , and that Π_{phys} shows partial nonlinear improvement but does not achieve a topology-independent within-profile collapse.

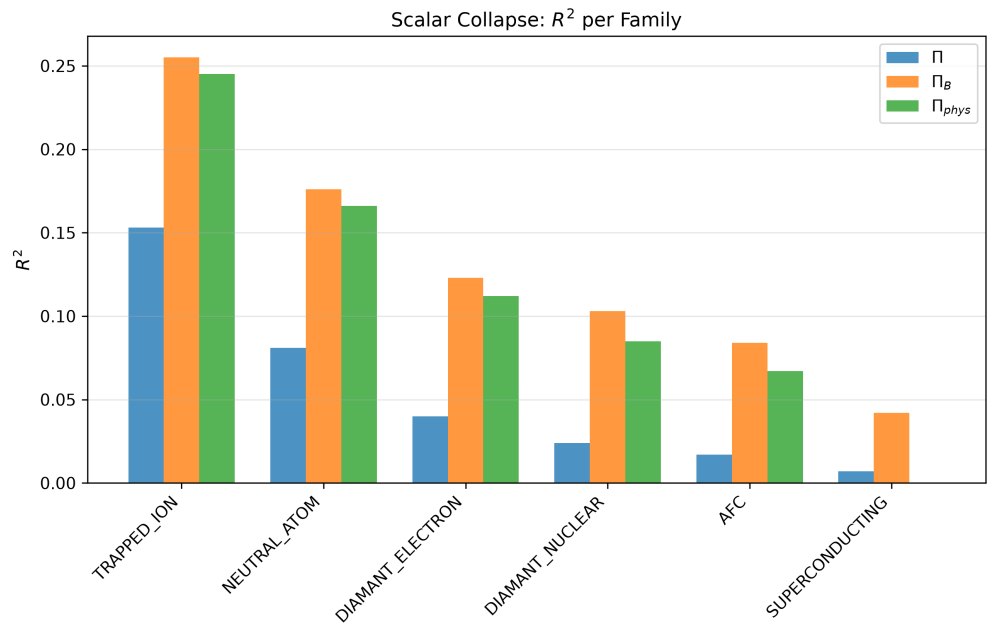


Figure 6. R^2 per family for Π , Π_B , and Π_{phys} . No scalar exceeds $R^2 = 0.26$ in any family.

Table 8. Nonlinear scalar-collapse test (pooled). R^2_{lin} : linear log-transformed. R^2_{quad} : quadratic log-transformed. ρ_s : Spearman rank correlation.

Scalar	R^2_{lin}	R^2_{quad}	ρ_s
Π	0.025	0.029	−0.118
Π^*	0.042	0.045	−0.169
Π_B	0.052	0.058	−0.214
Π_{phys}	0.177	0.379	−0.721

8. Retrieval-Window Descriptor and Limits of Analytical Sizing

The four scalar candidates above fail to predict m_c without circularity. We now examine the retrieval-window parameter $\chi_r = \tau_r p_{\text{gen}} / t_{\text{attempt}}$ (defined in Eq. 1), where τ_r is the retrieval decay time, as a physical descriptor of the operating regime. We present an independent-channel reference calculation, discuss its physical interpretation, and identify the limits of analytical sizing. No predictive accuracy or invariance rate is asserted for the corrected χ_r in this revision; the calculation below is an illustrative reference, not a validated predictor.

8.1. Independent-Channel Reference Calculation

For a single link, the number of link attempts within one retrieval lifetime is

$$K = \left\lfloor \frac{\tau_r}{t_{\text{attempt}}} \right\rfloor, \quad (14)$$

and the probability of at least one successful generation within the retrieval lifetime is

$$p_w = 1 - (1 - p_{\text{gen}})^K. \quad (15)$$

The memory budget needed to achieve target availability α on a single link is

$$m_{\text{single}} \approx \frac{\ln(1 - \alpha)}{\ln(1 - p_w)}. \quad (16)$$

Table 9 compares the single-link reference calculation to the observed median m_c across the four primary profiles.

Table 9. Single-link reference calculation vs. observed median m_c at $\alpha = 0.95$ (per-node budget). Observations are $2\text{--}3\times$ higher than the single-link reference, consistent with multi-hop paths requiring simultaneous link success. The reference is illustrative, not a validated predictor.

Profile	χ_r	p_w	Reference m	Observed median
Trapped Ion	50	≈ 1	1	4
Neutral Atom	0.5	0.39	6	16
Diamond (Electron)	0.22	0.20	14	32
AFC	0.025	0.025	120	64

The single-link reference underestimates the observed m_c by a factor of $2\text{--}3$. This is expected: in multi-hop networks, all h links along a path must succeed, so the effective path-success probability is $p_{\text{path}} = p_w^h$, which is much smaller than p_w and therefore requires more memory budget.

8.2. Multi-Hop Reference and AFC Illustrative Example

For a path of h hops, the memory threshold becomes

$$m_{\text{multi}}(h) \approx \frac{\ln(1 - \alpha)}{\ln(1 - p_w^h)}. \quad (17)$$

As an illustrative reference, consider the AFC profile ($\tau_r = 0.01$ s, $p_{\text{gen}} = 0.005$, $t_{\text{attempt}} = 10^{-3}$ s) on a 2-hop path at $\alpha = 0.95$:

- $K = \lfloor 0.01/0.001 \rfloor = 10$ attempts per retrieval lifetime
- $p_w = 1 - (1 - 0.005)^{10} \approx 0.0489$
- $p_{\text{path}} = p_w^2 \approx 0.00239$
- $m_{\text{multi}}(2) = \ln(0.05) / \ln(1 - 0.00239) \approx 1252$ channels

This ~ 1252 -channel figure is an illustrative reference for the coincidence-limited regime, not a validated prediction of the observed m_c . The observed AFC median $m_c \geq 64$ (Table 5) is lower than this reference, which does not account for topology-specific path lengths, traffic, or contention effects captured in the simulation.

8.3. Physical Interpretation

The χ_r values cleanly separate the four primary profiles into two regimes (Table 1):

- **Queue-limited** ($\chi_r \gg 1$, Trapped Ion $\chi_r = 50$): Many link successes occur within the memory lifetime, so the bottleneck is queueing and contention, not coincidence. Memory demand is governed by Little's law.
- **Coincidence-limited** ($\chi_r \ll 1$, AFC $\chi_r = 0.025$, Diamond Electron $\chi_r = 0.22$): Link success within the memory lifetime is rare. Memory demand is governed by the coincidence probability p_w , and m_c is large.
- **Transitional** ($\chi_r \sim 1$, Neutral Atom $\chi_r = 0.5$): Both mechanisms contribute.

This regime separation is consistent with the per-family m_c distribution (Table 5): Trapped Ion (queue-limited) has median $m_c = 4$, while AFC (coincidence-limited) has median $m_c \geq 64$. The retrieval-window descriptor thus provides a physical interpretation for the regime-dependent threshold that the four scalar candidates could not capture without circularity.

No predictive accuracy or invariance rate is asserted. The independent-channel approximation does not account for topology, traffic, contention, or interface effects. No

invariance test, factorial variance attribution, or benchmark comparison is reported for the corrected χ_r in this revision. The descriptor is offered as a physical timescale interpretation, not as a network memory-sizing law.

9. Heterogeneous Memory Placement

We resolve an apparent contradiction in earlier exploratory results with a controlled experiment over 6 topologies, 2 network sizes, 3 traffic levels, 3 coupling rules, and 5 seeds. Four mixed-placement strategies are evaluated at both high-quality fractions, while the two homogeneous controls (all-high and all-low) are evaluated once per graph/traffic/coupling condition. This gives 5,400 nominal runs. Ninety runs (1.7%) are skipped due to disconnected RGG graphs at $N = 20$, leaving 5,310 actual runs.

Definitions: *Hub* nodes are the top- k nodes by degree centrality. *Bottleneck* nodes are the top- k nodes by betweenness centrality. *Peripheral* nodes are the bottom- k nodes by degree centrality. In a star topology, the hub and bottleneck nodes coincide.

Result: Hubs (mean $A = 0.0815$) and bottlenecks (mean $A = 0.0766$) outperform peripheral placement (mean $A = 0.0323$). The strategy ranking is robust across coupling rules (Kendall $\tau = 1.0$ for all pairs of {min, sqrt, product}).

Paired comparison (hub vs bottleneck): Across 1,062 matched pairs, the median paired difference is 0 (558 ties, 322 hub wins, 182 bottleneck wins). A Wilcoxon signed-rank test is significant ($p < 0.001$) due to the asymmetry in non-tie outcomes, but the effect size is small (mean difference = 0.005, Cohen's $d = 0.19$). The Wilcoxon test uses the standard *wilcox* zero method, which discards the 558 zero-difference pairs before ranking, leaving 504 non-zero pairs; the p -value is computed via the normal approximation. A two-one-sided-tests (TOST) equivalence procedure with margin $\delta = 0.02$ confirms equivalence ($p_{\text{TOST}} < 0.001$), indicating that the mean paired difference is within ± 0.02 on the A_{req} scale. We conclude that hub and bottleneck placement are *statistically distinguishable but practically equivalent* in the tested configurations, supported by both the small effect size and the formal TOST result.

Per-topology ranking: The ranking is not perfectly consistent across topologies. In chain, peripheral outperforms bottleneck; in lattice, bottleneck outperforms hub. The overall ranking (hub > bottleneck > random > peripheral) holds in 4 of 6 topologies. Figure 7 shows the heterogeneous placement results across coupling rules.

9.1. Interface-Coupling Model Sensitivity

The three coupling rules (min, $\sqrt{\eta_i \eta_j}$, $\eta_i \eta_j$) are alternative model assumptions used to test sensitivity to the physical interface between heterogeneous memories at the two endpoints of a link. They produce different absolute performance levels but preserve the ranking of placement strategies. The min and $\sqrt{\cdot}$ rules produce similar absolute levels (within 2%), while the product rule lowers absolute A_{req} by 18–25% depending on strategy. These coupling rules are explicit modeling assumptions and are not native concepts in the external frameworks used in the cross-implementation study; they have not been independently validated. This confirms that heterogeneous-memory conclusions require explicit interface modeling.

9.2. Gap Between Mixed and All-High Placement

A notable finding is the large gap between any mixed placement and the all-high baseline: mixed strategies recover only 8–18% of the all-high performance. This gap quantifies the cost of heterogeneous hardware and is a distinct result from the ranking of mixed strategies.

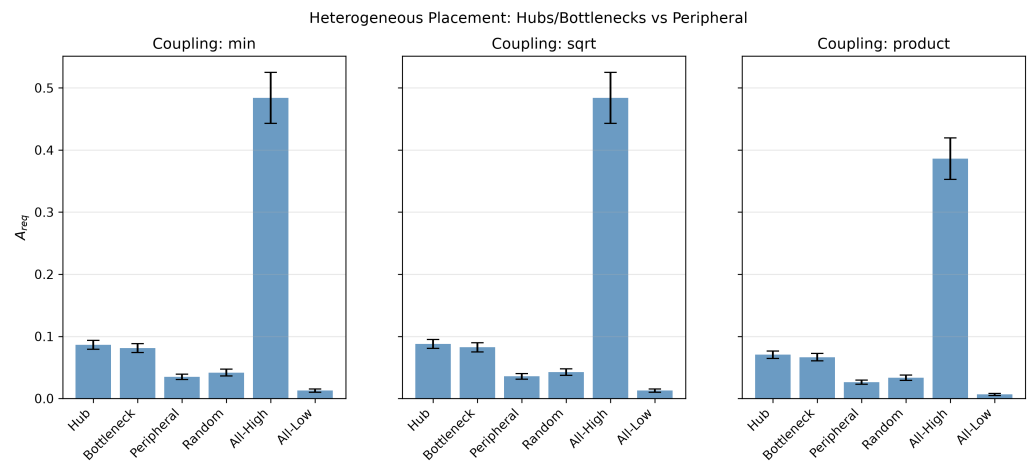


Figure 7. Heterogeneous placement results across three coupling rules. Aggregated across the tested configurations, hub and bottleneck placement outperform peripheral placement, although the per-topology ranking is not universal.

10. Global Sensitivity Analysis

10.1. Permutation Importance Cross-Check

As an independent sensitivity cross-check, we compute permutation importance using a Random Forest surrogate trained on the 67,584 A08 samples. Permutation importance measures which variables the surrogate is most dependent on for its predictions; it does not directly measure causal physical importance. The surrogate’s held-out R^2 and MAE are reported alongside the importance values to contextualize their reliability (Table 10).

Table 10. Random Forest surrogate performance on held-out data (20% test split, 200 trees). Per-topology R^2 is high; the lower overall R^2 reflects topology as a major driver not included as a feature.

Topology	n	R^2	MAE
All (pooled)	67,584	0.423	0.142
Chain	22,528	0.944	0.015
Star	22,528	0.972	0.019
RGG	22,528	0.974	0.023

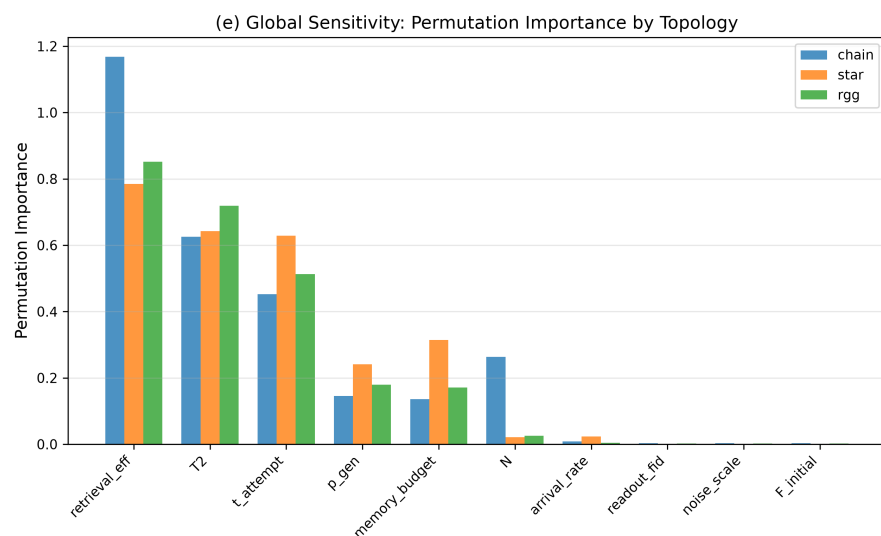
The per-topology R^2 values (0.94–0.97) indicate that the surrogate captures the within-topology input–output mapping well, so the permutation importance values are reliable as *predictive* associations. The pooled $R^2 = 0.42$ confirms that topology is a major driver of performance that is not captured by the 10 varied parameters alone. Table 11 lists the parameter ranges, and Figure 8 shows the permutation importance by topology.

Result: Retrieval efficiency is consistently ranked as the most informative predictor by the Random Forest surrogate across all three tested topologies (permutation importance 0.78–1.15), followed by T_2 (0.61–0.67) and t_{attempt} (0.40–0.59). The qualitative surrogate ranking is reproducible. The qualitative ranking is reproducible across topologies.

Caveat: Retrieval efficiency η_r enters the success probability multiplicatively, so its high importance is partly structural rather than purely physical. The importance of η_r should be interpreted as “most informative for the surrogate’s predictions,” not as “causally dominant in the physical system.” The broad range of η_r (0.30–0.99) also contributes to its high ranking; a narrower range would likely reduce its importance. The feature-importance ordering has not yet been evaluated across independent simulation models.

Table 11. Parameter ranges for the sensitivity analysis. Log-scaled parameters are sampled uniformly in log-space.

Parameter	Min	Max	Scale
η_r (retrieval eff.)	0.30	0.99	linear
F_{ro} (readout fid.)	0.90	0.999	linear
T_2	0.001 s	10 s	log
p_{gen}	0.001	0.05	log
t_{attempt}	10^{-5} s	10^{-2} s	log
m (memory budget)	1	8	discrete
λ (arrival rate)	1.0	100.0	log
noise scale	0.1	10.0	linear
N (network size)	10	50	discrete
F_{initial}	0.85	0.99	linear

**Figure 8.** Permutation importance by topology. Retrieval efficiency is consistently the most important parameter across all three topologies, followed by T_2 and t_{attempt} .

Why T_2 ranks high in the surrogate. The permutation-importance analysis ranks T_2 as the second-most informative predictor (0.61–0.67). In the atlas dataset, T_2 and τ_r are perfectly collinear across the six memory families (Spearman $\rho = 1.0$). The surrogate therefore uses T_2 as a proxy for τ_r ; permuting T_2 destroys the proxy and degrades predictions. The surrogate’s high ranking for T_2 is therefore a consequence of collinearity in the training data, not necessarily of independent physical causation. The three metrics— A_{req} (delivery without fidelity), A_{qual} (delivery with fidelity threshold), and fidelity at delivery—are affected differently: T_2 affects fidelity and A_{qual} through decoherence; τ_r affects all three through retrieval decay. A surrogate trained on A_{qual} or fidelity would rank T_2 highly for a direct physical reason; a surrogate trained on A_{req} may rank T_2 highly because it proxies for τ_r .

11. Design Implications

1. **The χ_r descriptor provides a physical timescale interpretation.** The coincidence-window parameter $\chi_r = \tau_r p_{\text{gen}} / t_{\text{attempt}}$ (where τ_r is the retrieval decay time) separates queue-limited ($\chi_r \gg 1$) from coincidence-limited ($\chi_r \ll 1$) regimes and is consistent with the observed per-family m_c distribution. No predictive accuracy or invariance rate is asserted for χ_r in this revision. For network planning, χ_r can be

- computed from device parameters as a physical descriptor, but it does not establish a network memory-sizing law.
2. **Memory selection cannot be reduced to coherence time alone.** Retrieval efficiency, coherence, and waiting-time-induced decoherence interact; retrieval efficiency and coherence time rank highly in the tested surrogate analyses, while memory-profile choice produces large performance differences. This is a structural conclusion; the specific importance values are model-specific estimates.
 3. **Retrieval efficiency, coherence, and waiting time interact.** The cross-implementation check reproduced the ordering of the evaluated memory profiles under the fidelity-qualified metric, confirming that the family ordering is a structural finding within the tested model class.
 4. **Network topology changes the value of a given memory resource.** The confirmatory subset exhibits a topology effect (RGG mean $A_{\text{req}} = 0.233$ vs chain mean 0.082), though this comparison is confounded with path length and total memory (topologies with more nodes receive more total memory under the per-node budget definition). The RGG > star > chain ordering was reproduced in the cross-implementation check. The tested scalar scaling hypotheses remain topology-dependent, so mean path-length corrections alone are insufficient to collapse the observed variation.
 5. **Memory overprovisioning thresholds are regime-specific.** A single m_c value does not describe the tested configurations; capacity planning must account for memory profile, topology, and offered load. The m_c values reported here are model-specific quantitative estimates of the WestQuant simulation; the structural conclusion that m_c is regime-dependent is robust.
 6. **High-quality resources should preferentially occupy structurally important nodes under the tested placement model.** Assigning high-quality memories to hubs or traffic-bottleneck nodes outperforms peripheral placement, and the strategy ordering is unchanged across the three coupling rules. This is a structural conclusion; the placement result has not yet been cross-implemented.
 7. **Interface assumptions can materially alter absolute performance.** Coupling-rule assumptions affect absolute performance by 18–25%. The interface-coupling rules are explicit modeling assumptions and have not been independently validated. Heterogeneous-memory conclusions require explicit interface modeling.

The representative memory profiles used here are parameter bundles, not complete realized hardware stacks; the design messages above are qualitative rather than device-certification-level.

12. Limitations

1. **Cross-implementation rather than full native-protocol validation.** In addition to analytical component checks, selected conclusions were tested through harmonized implementations executed on SeQUeNCe and SimQN event frameworks. The two external adapters share a common harmonized model and therefore do not constitute two independent physical-model replications. Native SeQUeNCe/SimQN protocols, NetSquid, and QuISP were not included.
2. **Absolute-value model dependence.** Cross-implementation comparisons indicate that absolute performance values are more implementation-sensitive than the tested topology and memory-profile orderings.
3. **Critical memory capacity.** Numerical critical-memory thresholds have not yet been cross-validated using the full $m = \{1, 2, 4, 8, 16, 32, 64\}$ sweep.
4. **Heterogeneous placement.** The heterogeneous placement result has not yet been cross-implemented.

5. **Interface-coupling rules.** The interface-coupling rules are explicit modeling assumptions and have not been independently validated. 586
587
6. **Metric semantics.** Completion and latency metrics are sensitive to queueing, censoring, and timing definitions; future native-protocol comparisons must harmonize these definitions before quantitative comparison. 588
589
590
7. **Scalar collapse.** The scalar-collapse result has not yet been independently cross-implemented. 591
592
8. **Sobol bootstrap bug.** The Saltelli bootstrap confidence intervals were found to be unreliable due to a structural resampling bug (resampling individual rows instead of Saltelli blocks). Sobol indices are therefore not reported as quantitative results; fixing the bootstrap and rerunning is future work. We rely on permutation importance for qualitative ranking. 593
594
595
596
597
9. **Representative memory parameters.** Memory families use representative parameter bundles, not complete hardware-stack comparisons. Real devices have additional noise sources. 598
599
600
10. **Uniform traffic in main atlas.** Most atlas experiments use uniform traffic. Traffic-matrix support exists but is not exercised in the main atlas. 601
602
11. **No purification or multiplexing.** The simulator does not model entanglement purification or spatial multiplexing. 603
604
12. **Limited network sizes.** Experiments cover $N \in [10, 50]$. Scaling to $N \geq 100$ is future work. 605
606
13. **Routing policy.** We use shortest-hop routing. Other policies may change conclusions. 607
14. **No dynamic traffic.** Traffic is Poisson with fixed rate. Time-varying traffic belongs to future work. 608
609
15. **Atlas and confirmatory grid completeness.** The atlas design contains 9,720 nominal cells; 9,360 completed runs are retained (360 missing, 3.7%). The confirmatory grid contains 9,000 cells; 8,730 completed simulations are retained (270 missing, 3.0%). Missing cells arise from disconnected random-geometric graphs and simulation failures and are not imputed. 610
611
612
613
614
16. **Topology fairness: per-node vs total-memory confound.** Because m is defined as the per-node budget, topologies with more nodes receive more total memory. Cross-topology comparisons therefore conflate topology structure with total resource allocation. A fixed-total-memory design would be needed to isolate the pure topology effect. 615
616
617
618
619
17. **Topology-dependent exclusion of no-path requests.** The metric A_{req} excludes no-path requests from the denominator. Because the fraction of no-path requests depends on topology (e.g., disconnected RGGs produce more no-path requests than fully connected chains), the effective denominator varies across topologies, potentially biasing cross-topology comparisons. 620
621
622
623
624
18. **Seed and request quantization.** The atlas uses 5 seeds and 100 requests per simulation. With only 100 requests, A_{req} is quantized to multiples of $1/100$, limiting resolution for small differences. The confirmatory dataset uses 20 seeds, which reduces but does not eliminate this quantization effect. 625
626
627
628
19. **m_c relative to observed maximum.** The threshold m_c is defined relative to $A_{\text{max}}^{\text{obs}}$, the maximum on the observed budget grid, not an asymptotic limit. If the budget-response curve has not plateaued, m_c underestimates the budget needed to reach a fixed fraction of the true asymptotic performance. This is a fundamental limitation of grid-based threshold estimation. 629
630
631
632
633
20. **RGG disconnection selection bias.** Disconnected random-geometric graphs are excluded from the analysis (contributing to the 360 missing atlas cells and 270 missing 634
635

- confirmatory cells). This creates a selection bias: the surviving RGG instances are those with sufficient edge density to remain connected, which may overstate RGG performance relative to a random draw from the RGG ensemble.
21. **Readout-model discrepancy.** The frozen datasets were generated with a multiplicative readout model ($F_{\text{out}} = F_{ro} \cdot F$), while the current codebase uses a depolarizing model ($F_{\text{out}} = F_{ro} \cdot F + (1 - F_{ro}) \cdot 0.25$). The difference is 1–2.5% in fidelity (systematic: the depolarizing model yields higher fidelity than the multiplicative model for $F_{ro} < 1$ and $F > 0.25$). This does not affect the retrieval-window descriptor χ_r because χ_r depends on τ_r and generation parameters, not readout. Regenerating the frozen datasets with the depolarizing model is future work.
 22. **Run length at high traffic.** With $n_{\text{requests}} = 100$ and $\lambda = 50$, the observation window is approximately 2 s, which may not reach full stationarity for queueing dynamics at high load. Results at $\lambda = 50$ should be interpreted with this caveat.

13. Conclusion

Memory capacity requirements in the tested quantum-network model depend on memory profile, topology, and traffic. The finite-grid threshold reaches the upper tested budget in 35.5% of the reported regimes, and it must be distinguished from an asymptotic saturation capacity. The tested scalar reductions leave substantial topology-dependent variation. Correctly using retrieval decay time in χ_r provides a physical timescale descriptor but does not establish a universal sizing law. Heterogeneous placement favors hubs and betweenness-based nodes in aggregate, while topology-specific exceptions and coupling-dependent absolute performance limit generalization. These findings support explicit, model-specific capacity planning and targeted validation of the decisions to be deployed.

Data Availability

The manuscript and a repository scaffold are prepared for a dedicated public reproducibility release. The raw simulation datasets, frozen simulator, dependency lockfile, complete seed/configuration ledger, and figure-generation scripts have not yet been deposited in that release. The development repository is private. This revision therefore does not claim that the empirical results can already be reproduced from a public archive. A public release URL, commit identifier, checksums, and archival DOI should identify the completed deposit before submission.

Relationship to Companion Work

A companion methods manuscript, “Assessing Decision Reliability in Discrete-Event Simulation: A Quantum-Network Case Study,” studies stochastic decision stability, metric semantics, and model sensitivity. The present article focuses on memory-profile, capacity, topology, and placement results. Shared simulator infrastructure and overlapping validation results should be disclosed to both editors.

Author Contributions: Conceptualization, D.V.; methodology, D.V.; software, D.V.; validation, D.V.; formal analysis, D.V.; investigation, D.V.; resources, D.V.; data curation, D.V.; writing—original draft preparation, D.V.; writing—review and editing, D.V.; visualization, D.V.; supervision, D.V.; project administration, D.V.; funding acquisition, D.V.

Funding: This research received no external funding. Computational resources were provided by the author’s personal hardware. No GPU or compute-cluster grants were used.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The manuscript and a repository scaffold are prepared for a dedicated public reproducibility release. The raw simulation datasets, frozen simulator, dependency lockfile, complete seed/configuration ledger, and figure-generation scripts have not yet been deposited in that release. The development repository is private. A public release URL, commit identifier, checksums, and archival DOI should identify the completed deposit before submission.

Acknowledgments: The author acknowledges the use of open-source quantum-network simulation frameworks SeQUeNCe and SimQN in the cross-implementation robustness check. AI-assisted tools (Anthropic Claude, OpenAI ChatGPT, and Cognition Devin) were used during the preparation of this work, including assistance with code generation, manuscript organization, and editorial revision. The author takes full responsibility for the work.

Conflicts of Interest: The author is the founder of Vesterlund Ventures and WestQuant Open. The simulator used in this study (WestQuant) was developed by the author. This creates a potential conflict of interest insofar as the simulator's design choices may influence the results. The cross-implementation robustness check partially mitigates this concern by testing whether selected conclusions survive an independent implementation pathway, but does not eliminate it.

Abbreviations

The following abbreviations are used in this manuscript:

T_2	Coherence time
τ_r	Retrieval decay time
η_r	Retrieval efficiency
F_{ro}	Readout fidelity
χ_r	Retrieval-window parameter ($\tau_r p_{gen} / t_{attempt}$)
m_c	Critical memory capacity
A_{req}	Requirement-qualified availability
A_{qual}	Quality-qualified availability
RGg	Random geometric graph
AFC	Atomic frequency comb
RF	Random Forest

1. S. Wehner, D. Elkouss, and R. Hanson, Quantum internet: A vision for the road ahead, *Science* **362**, eaam9288 (2018).
2. M. Razavi, M. Piani, and N. Lütkenhaus, Quantum repeaters with imperfect memories: Cost and scalability, *Phys. Rev. A* **80**, 032301 (2009).
3. N. Sangouard, C. Simon, H. de Riedmatten, and N. Gisin, Quantum repeaters based on atomic ensembles and linear optics, *Rev. Mod. Phys.* **83**, 33 (2011).
4. L. Kamin, E. Shchukin, F. Schmidt, and P. van Loock, Exact rate analysis for quantum repeaters with imperfect memories and entanglement swapping as soon as possible, *Phys. Rev. Research* **5**, 023086 (2023).
5. M. Chehimi, S. Pouryousef, N. K. Panigrahy, D. Towsley, and W. Saad, Scaling limits of quantum repeater networks, arXiv:2305.08696 (2023).
6. X. Wu, A. Kolar, J. Chung, D. Jin, T. Zhong, R. Kettimuthu, and M. Suchara, SeQUeNCe: A customizable discrete-event simulator of quantum networks, *Quantum Sci. Technol.* **6**, 045022 (2021).
7. T. Coopmans, R. Knegjens, A. Dahlberg, D. Ferrari, J. Rabbie, and S. Wehner, NetSquid, a network simulator for quantum information using discrete events, *Commun. Phys.* **4**, 164 (2021).
8. R. Satoh, M. Hajdušek, N. Benchasattabuse, S. Suzuki, R. Van Meter, and H. Takesue, QuISP: A quantum internet simulation package, in 2022 *IEEE Int. Conf. on Quantum Computing and Engineering (QCE)*, pp. 353–364 (2022).
9. J. Chung, M. Hajdušek, N. Benchasattabuse, A. Kolar, A. Singal, K. S. Soon, K. Teramoto, A. Zang, R. Kettimuthu, and R. Van Meter, Cross-validating quantum network simulators, arXiv:2504.01290 (2025).

10. K. S. Soon, N. Benchasattabuse, M. Hajdušek, K. Teramoto, S. Nagayama, and R. Van Meter, Performance of quantum networks using heterogeneous link architectures, in *2024 IEEE Int. Conf. on Quantum Computing and Engineering (QCE)*, pp. 1914–1923 (2024), arXiv:2405.09862. 723–725
11. S. Sun, D. Bhatti, S. Gao, D. Elkouss, and H. Takahashi, Hybrid quantum repeater chains with atom-based quantum processing units and quantum memory multiplexers, arXiv:2512.21655 (2025). 726–728
12. P. Cussenot, B. Grivet, B. P. Lanyon, T. E. Northup, H. de Riedmatten, A. S. Sørensen, and N. Sangouard, Uniting quantum processing nodes of cavity-coupled ions with rare-earth quantum repeaters using single-photon pulse shaping based on atomic frequency comb, *Phys. Rev. Lett.* **135**, 240803 (2025), arXiv:2501.18704. 729–732
13. K. Ooi, G. Sutcliffe, and M. Beghelli, Impact of topology on multipartite entanglement distribution protocols in quantum networks, *J. Opt. Commun. Netw.* **18**, 1054 (2026). 733–734
14. M. Ozaydin et al., Optimal entanglement distribution within a multi-ring topology, *Sci. Rep.* (2025). 735–736
15. G. Vardoyan, E. van Milligen, S. Guha, S. Wehner, and D. Towsley, On the bipartite entanglement capacity of quantum networks, *IEEE Trans. Quantum Eng.* **5**, 1 (2024), arXiv:2307.04477. 737–738
16. S. Pouryoucef, H. Shapourian, A. Shabani, R. Kompella, and D. Towsley, Resource placement for rate and fidelity maximization in quantum networks, arXiv:2308.16264 (2024). 739–740
17. R. Sripotchanart, W. Si, R. N. Calheiros, Q. Cao, and T. Qiu, A two-step linear programming approach for repeater placement in large-scale quantum networks, *Comput. Netw.* **254**, 110795 (2024). 741–743
18. T. Islam and E. Arslan, Scalable quantum repeater deployment modeling, arXiv:2305.09855 (2023). 744–745
19. S. Pirandola, End-to-end capacities of a quantum communication network, *Commun. Phys.* **2**, 51 (2019). 746–747
20. M. Takeoka, S. Guha, and M. M. Wilde, Fundamental rate-loss tradeoff for optical quantum communication, *Nat. Commun.* **7**, 13523 (2016). 748–749
21. T. Vasantam and D. Towsley, Stability analysis of a quantum network with max-weight scheduling, arXiv:2106.00831 (2021). 750–751
22. C. Langer, R. Ozeri, J. D. Jost, J. Chiaverini, B. DeMarco, A. Ben-Kish, R. B. Blakestad, J. Britton, D. B. Hume, W. M. Itano, D. Leibfried, R. Reichle, T. Rosenband, T. Schaetz, P. O. Schmidt, and D. J. Wineland, Long-lived qubit memory using atomic ions, *Phys. Rev. Lett.* **95**, 060502 (2005). 752–754
23. T. P. Harty, D. T. C. Allcock, C. J. Ballance, L. Guidoni, H. A. Janacek, N. M. Linke, D. N. Stacey, and D. M. Lucas, High-fidelity preparation, gates, memory, and readout of a trapped-ion quantum bit, *Phys. Rev. Lett.* **113**, 220501 (2014). 755–757
24. Y. Wang et al., Coherent control of the fine-structure qubit in a single alkaline-earth atom, *Phys. Rev. Lett.* **132**, 150606 (2024). 758–759
25. M. Hosseini, B. M. Sparkes, G. Campbell, P. K. Lam, and B. C. Buchler, High efficiency coherent optical memory with warm rubidium vapour, *Nat. Commun.* **2**, 174 (2011). 760–761
26. Y.-F. Hsiao, P.-J. Tsai, H.-S. Chen, S.-X. Lin, C.-C. Hung, C.-H. Lee, Y.-H. Chen, Y.-F. Chen, I. A. Yu, and Y.-C. Chen, Highly efficient coherent optical memory based on electromagnetically induced transparency, *Phys. Rev. Lett.* **120**, 183602 (2018). 762–764
27. G. Balasubramanian et al., Ultralong spin coherence time in isotopically engineered diamond, *Nat. Mater.* **8**, 383 (2009). 765–766
28. M. Steiner, P. Neumann, J. Beck, F. Jelezko, and J. Wrachtrup, Universal enhancement of the optical readout fidelity of single electron spins at nitrogen-vacancy centers in diamond, *Phys. Rev. B* **81**, 035205 (2010). 767–769
29. C. E. Bradley et al., Robust quantum-network memory based on spin qubits in isotopically engineered diamond, *npj Quantum Inf.* **8**, 122 (2022). 770–771
30. P. C. Maurer, G. Kucsko, C. Latta, L. Jiang, N. Y. Yao, S. D. Bennett, F. Pastawski, D. Hunger, N. Chisholm, M. Markham, D. J. Twitchen, J. I. Cirac, and M. D. Lukin, Room-temperature quantum bit memory exceeding one second, *Science* **336**, 1283 (2012). 772–774
31. A. Ortu, A. Holzäpfel, J. Etesse, and M. Afzelius, Storage of photonic time-bin qubits for up to 20 ms in a rare-earth doped crystal, *npj Quantum Inf.* **8**, 29 (2022). 775–776

32. C. Laplane, P. Jobez, J. Etesse, N. Gisin, and M. Afzelius, Multimode and long-lived quantum correlations between photons and spins in a crystal, *Phys. Rev. Lett.* **118**, 210501 (2017). 777
33. M. Reagor et al., Quantum memory with millisecond coherence in circuit QED, *Phys. Rev. B* **94**, 014506 (2016). 778
34. See Supplemental Material at <https://github.com/VesterlundCoder/quantum-network> (supplemental material) for full parameter provenance (Table S1), complete m_c tables (Table S2), cross-implementation validation appendix (Table S6), and nonlinear scalar-collapse details (Table S7). Sobol sensitivity tables (formerly S3–S5) have been removed due to the bootstrap bug described in Sec. 10. 779