This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/TQE.XXXX.DOINumber

Q-REFINE: A Quantum Datacenter Design Optimization Framework

CHENLIANG TIAN[1], RAJ JAIN[1], (LIFE FELLOW, IEEE), REZA NEJABATI[2], ENEET KAUR[2], RAMANA KOMPELLA[2], AIMAN ERBAD[3], and MOHAMED ABDALLAH[4]

[1]Department of Computer Science and Engineering, Washington University in St. Louis, St. Louis, MO 63130 USA
[2]Quantum Lab, Cisco Systems, Inc., Santa Monica, CA 90401 USA
[3]College of Engineering, Qatar University, Doha, Qatar
[4]College of Science and Engineering, Hamad Bin Khalifa University, Doha, Qatar

Corresponding author: Chenliang Tian (e-mail: chenliang.t@wustl.edu).

This work was supported in part by Cisco University Research Grant #6138 and the Qatar Research, Development, and Innovation (QRDI) Academic Research Grant #ARG02-0415-240191. The statements made here are solely the responsibility of the authors.

ABSTRACT Quantum datacenter (QDC) architectures must support communication among quantum processing units (QPUs) under limited communication qubits, optical switching and measurement resources, probabilistic entanglement generation, and physical-layer noise. This paper presents Q-REFINE, a framework for Quantum Resource Exploration via Factorial & Iterative N-factor Experiments. Its measure is the completion rate, the rate at which requests are served with at least a threshold fidelity. For a given QDC topology, Q-REFINE asks which physical-layer factors and their combinations change this rate most, and over what range improving each factor still pays off. Q-REFINE combines a discrete-event simulator, controllable inter-QPU request workloads, topology instances with explicit switches and measurement units, and a design of experiments (DOE). We apply it to Fat-tree, Clos, QFly, BCube and DCell. Load selection fixes, with every factor at its best level, the smallest concurrency limit whose completion rate is within 5% of the best while at most 5% of requests go unserved. At that limit, a full-factorial DOE finds which factor combinations serve any request and how much each factor raises the rate, and one-factor-at-a-time sweeps show over what range each improvement pays off. Emitted-pair fidelity, Hong–Ou–Mandel visibility and memory coherence time jointly decide whether any request is served. The emission efficiency raises the rate most, 8.4–9.5 times between its two DOE levels, and a higher attempt rate pays off only with a short communication-qubit cooldown time. Although we use a particular workload, performance metric, and configurations, the framework applies to any workload, metric(s), or configurations.

INDEX TERMS Quantum Datacenter Design Optimization, Quantum Factorial Analysis and Configuration Tuning, Quantum Multi-factor Optimization and Design, Quantum Resource Analysis via Factorial Testing, Quantum networks, Quantum Datacenter sensitivity analysis.

I. INTRODUCTION

Quantum datacenters (QDCs) are a systems-level approach to scaling quantum computation beyond a single quantum processing unit (QPU) [1]–[3]. A QDC connects multiple QPUs via an optical interconnect, enabling entanglement to be established between processors for non-local operations. In this setting, performance depends not only on the quality of individual QPUs, but also on how the interconnect is built and operated, because the topology, optical components, communication qubits, switching and measurement resources, and physical-layer parameters all shape the rate and fidelity of inter-QPU communication.

This design space is difficult to reason about because architectural and physical effects are coupled. Changing the topology changes the number of switches, optical links, measurement resources, possible relay points, and shared bottlenecks encountered by a request. Changing physical-layer parameters such as loss, emission efficiency, attempt rate, memory coherence, and dark-count rate changes the probability, timing, and final fidelity of entanglement generation. A designer therefore needs, for each topology, the answers to two questions, namely which physical-layer

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

factors and combinations of factors most change the rate at which requests are served above a fidelity threshold, and over what range improving each factor still pays off.

Prior work has examined several pieces of this problem. QDC architecture studies compare topologies such as Clos, Fat-tree, BCube, QFly, DCell, and related designs [1]–[6], [7], [8]. Scheduling and compilation studies examine how distributed quantum programs can be mapped and coordinated across QPUs [9], [10]–[13]. Quantum-network simulators provide detailed models of noisy links, communication protocols, and resource management [14]–[19]. These works show that both topology and physical-layer assumptions matter. Still, none of them answers these two questions for a QDC topology, where many requests contend for shared switch ports and Bell-state measurement (BSM) units under a controlled workload.

This paper addresses that gap with Q-REFINE, a framework for Quantum Resource Exploration via Factorial & Iterative N-factor Experiments. We apply Q-REFINE to five topology families, Fat-tree, Clos, QFly, BCube and DCell, and answer both questions for each family at a load selected for it. Each family keeps the switches, links and relay QPUs its construction requires, because these belong to the topology, so the results describe each family on its own and do not rank them.

The results of any performance analysis depend on three key decisions by the analyst, namely the configuration, the workload and the metric. Any change to one of these can significantly affect the conclusions. Also, there are no standard configurations, workloads, or metrics. So each of these can be questioned. Therefore, our goal here is not to sell particular conclusions but to show a methodology or framework that can be applied to any other configuration, workload, or set of metrics.

In summary, this paper makes five contributions to architecture-level QDC evaluation:

1. We define a three-stage design-optimization workflow (contributions 3–5) that applies to each QDC topology family under the same endpoint pool, workload, BSM total and factor ranges (Section III-C).
2. We implement a discrete-event simulator of how a QDC admits requests, reserves their paths, and generates, stores and joins entanglement (Section III). Unlike the tools we review, it couples a link model whose cost grows far more slowly than the number of attempts to a switch fabric whose ports and BSM units the requests contend for (Section II-C).
3. Because the concurrency limit, the number of requests that may compete for resources at once, sets how strongly requests contend, the first stage (load selection) sweeps it for each topology at the favorable corner, where every factor is at its favorable level, its best quality within the range studied, and selects for each topology the limit at which the completion rate stops rising.
4. At the selected concurrency limit, we conduct a 2^N full-factorial design of experiments (DOE) over the eleven physical-layer factors, each at a favorable and a less favorable level, to find which combinations serve any request and how much each factor and each interaction multiplies the completion rate where they do.
5. One-factor-at-a-time (OFAT) sweeps vary each factor over its range with every other factor at its favorable level, and show where along its range each factor stops raising the completion rate.

While DOE is quite common in medical and agricultural studies, its use in computer systems is rare. To our knowledge, a full-factorial DOE has not been used to attribute QDC interconnect performance to physical-layer factors and their interactions. For details of how to use DOE and its variations, see Jain [20].

The remainder of the paper is organized as follows. Section II reviews QDC architecture studies, execution and orchestration work, simulation frameworks, and the gap Q-REFINE addresses. Section III defines design optimization and the Q-REFINE framework, including its three stages, the topologies, the workload and the simulator's models. Section IV describes how the simulator is implemented, including its modules, one run as pseudocode, and why each run is reproducible. Section V defines the experimental design, including the main evaluation setting, the physical-layer factors and fidelity threshold, load selection, and the DOE and OFAT methods. Section VI presents the load-response and factor-sensitivity results. Section VII discusses design implications, limitations, and future extensions. Section VIII concludes the paper. Appendices A–G give the validation against event-level simulators, the link-model derivations, the evaluation settings, the simulator's modules, other sizes, the factor ranges with their sources and DOE levels, and additional results.

II. RELATED WORK

A. QUANTUM DATACENTER ARCHITECTURES AND TOPOLOGY STUDIES

At design time, QDC architecture studies define how QPUs, optical switches, measurement resources, and shared entanglement resources can be organized inside a datacenter. QDC architectures fall into two broad categories. In switch-centric architectures, every request is carried by a shared optical-switch fabric, and the QPUs act only as endpoints. In server-centric architectures, QPUs also act as relays that join adjacent links by entanglement swapping, so a request's path can pass through other QPUs.

Shapourian et al. [2] present a scalable architecture that separates switch-centric and server-centric designs and connects entanglement generation to network orchestration. Sakuma et al. [4] introduce Q-Fly, a Dragonfly-style optical interconnect that routes photons either within a local Bell-state analyzer pool or across low-diameter inter-group paths. Xin and Zhang [5] study a fat-tree architecture in which leaf

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

switches, swapping spine switches, and queue scheduling jointly manage entanglement requests and quantum memory.

Other architecture studies evaluate how topology choices affect communication and execution behavior under specific hardware assumptions. Pouryousef et al. [1] benchmark QFly, BCube, Clos, and Fat-tree under quantum-hardware constraints, including optical loss, coherence-limited retries, limited communication resources, and topology-induced contention. Cacciapuoti et al. [3] explain why QDC networks differ from classical datacenter networks, emphasizing entanglement generation, quantum memory, and distributed execution as core constraints. Together, these studies show that topology shapes the contention requests meet.

B. DISTRIBUTED EXECUTION AND ORCHESTRATION

Given a QDC architecture, another line of work studies how distributed quantum programs are partitioned, placed, and coordinated across QPUs. Communication-reduction methods address this problem before runtime execution begins. Kaur et al. [10] reduce Einstein–Podolsky–Rosen (EPR) pair demand by optimizing circuit partitioning across multiple QPUs with graph partitioning and integer programming. Burt et al. [11] generalize this view by treating gate teleportation and state teleportation within a unified graph-based framework. AutoComm [12] identifies burst communication patterns in distributed quantum programs and uses them to reduce communication cost and latency.

Runtime scheduling and orchestration studies then examine how inter-QPU communication is handled once execution begins. Pouryousef et al. [9] compare static and dynamic entanglement scheduling for remote gate execution under probabilistic entanglement generation, limited communication qubits, switch reconfiguration delay, and topology-induced contention. Zhang et al. [7], [8] develop SwitchQNet to connect circuit execution with reconfigurable switch control in switch-based QDC architectures. Chen et al. [13] study adaptive resource orchestration with centralized entanglement resources, shared memory, parallel entanglement attempts, and opportunistic caching. Yang et al. [6] formulate QDC scheduling with a mixed-integer linear programming model and propose the Least Resource First (LRF) scheduler and its group variant, Group LRF (GLRF), for DCell, showing that lightweight heuristics can reduce solver overhead while preserving scheduling quality.

These studies optimize how communication is reduced, scheduled, or coordinated for a given architecture or workload. They are complementary to the present study, which does not propose a new compiler or scheduler. Instead, we hold the rules for admitting, routing and reserving requests fixed and ask the two questions of Section I for each topology.

C. QUANTUM NETWORK AND DISTRIBUTED-QUANTUM SIMULATION TOOLS

Simulation tools provide another foundation for QDC evaluation. NetSquid [14] supports discrete-event simulation of quantum networks and modular quantum-computing systems, including noisy channels, decoherence, control protocols, and application-level behavior. SeQUeNCe [15] provides hardware, entanglement-management, resource-management, network-management, and application modules for quantum-network studies. QuISP [16] targets large-scale quantum-internet protocols in noisy and heterogeneous repeater networks. SimQN [17] provides network-layer simulation support for quantum key distribution protocols, entanglement distribution, routing, and resource allocation.

Distributed quantum computing simulators are moving closer to application execution. Ferrari and Amoretti [21] present a framework for simulating distributed quantum-computing systems and define QPU-utilization and quantum-network-utilization metrics for job scheduling. Zhang et al. [19] developed SimDisQ, an end-to-end distributed quantum-circuit simulator that connects circuit-level simulation, heterogeneous backends, noisy communication, and distributed execution. Avis et al. [22] come closest to our questions, deriving minimum hardware-parameter requirements for a repeater chain with NetSquid; they do not model many requests contending for a shared switch fabric.

These tools target detailed protocol and network behavior, job scheduling or end-to-end circuit execution. Our two questions (Section I) place three requirements on a simulator.

First, the design space is defined by parameters, not by platforms. Its ranges come from several hardware platforms, such as trapped ions and NV centers, so the link model must expose one set of parameters that each platform reaches by combining its own measured quantities, as Avis et al. do when they collect a node's hardware losses into one photon-detection probability [22]. We model every link with the double-click scheme, which all these platforms have demonstrated, so one link model with these parameters serves each of them. Results then read as requirements on parameters rather than as verdicts on platforms.

Second, the simulator must stay fast when attempts rarely succeed, because every combination of a factorial design is run many times. A link needs on average $1/P_{acc}$ attempts, where P_{acc} is the probability that one attempt heralds a pair, and P_{acc} can fall to about 3×10^{-8} at the least favorable emission efficiency reported (Appendix F), so a simulator that treats every attempt as an event would spend about 3×10^{7} events on one pair. The cost of a pair must therefore grow far more slowly than the number of attempts.

Third, the link model must be coupled to the QDC fabric, i.e., the optical switches and links that connect its QPUs, because the requests of a run contend for the fabric's switch ports and the BSM units at its switches while their links are generated and stored; this coupling, not the link model itself, is the simulator contribution of this work.

Table I compares the cited tools against these requirements; none meets all three. SeQUeNCe [15] and QuISP [16] simulate every attempt, or every photon, as an event, so

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

TABLE I. Cited simulators against the three requirements of Section II-C, and their access. MIM: memory–interface–memory link

Tool	Platform-independent double-click parameters (req. 1)	Cost per pair far below $1/P_{acc}$ events (req. 2)	QDC switch fabric (req. 3)	Access
SeQUeNCe [15]	yes	no; an event per attempt	not identified	open source
NetSquid [14] with netsquid-magic	yes	yes; analytic	not identified	snippets open; core restricted
QuISP [16]	yes (MIM link)	no; an event per photon	not identified	open source
SimQN [17]	no heralding model	not applicable	not identified	open source
QDC studies [1], [9], [7], [8]	no; fixed probability or latency	not applicable	BSM pools yes; ports no	[7], [8] open; [1], [9] not released
Q-REFINE	yes	yes; one geometric draw per generation round (Section IV-B)	yes	not released

their cost per pair grows as $1/P_{acc}$, whereas the netsquid-magic models that Avis et al. [22] use on NetSquid [14] sample many attempts at once. In the versions and public source we assessed, no general-purpose tool in Table I models the switch port budgets and per-switch BSM units that requests share, although an event-level framework such as NetSquid or SeQUeNCe could host such a model.

In the papers we assessed, and for [7], [8] also the public code, the QDC studies of Sections II-A and II-B do model shared per-switch BSM units [1], [9], [7], [8]. Each, however, takes a link as a fixed success probability or a fixed latency and tracks no fidelity, treating memory at most as a retry cutoff, and none treats switch ports as a resource that requests contend for. Like netsquid-magic, our simulator draws the number of attempts to the first herald once per round of attempts (Section IV-B).

Where our simulator's link and swap models overlap theirs, we checked its results against them, and we checked its model of shared communication qubits, BSM units and switch ports, which these tools do not include, separately (Appendix A).

D. GAP AND POSITIONING

These three lines of work leave open the two questions of Section I. Q-REFINE answers them for each topology at a load selected for it, with a full-factorial DOE [23], [20] and OFAT sweeps (Section III).

III. QDC DESIGN OPTIMIZATION FRAMEWORK

A. DESIGN OPTIMIZATION IN THIS STUDY

Once the topology of a QDC is chosen, its physical components, such as memories, optical switches, fibers and detectors, can each be built to a range of quality, and better quality costs more or is not yet available, so building every component at its best is rarely possible. Components interact, so improving one may pay off only when another is also good, and the completion rate may stop improving before a component reaches its best quality. These reasons motivate the two questions of Section I.

In this study, a design is one level of each physical-layer factor, such as emission efficiency or switch insertion loss, on a given topology family. A factor's favorable level is its best quality within the range studied, for example the highest emission efficiency or the lowest insertion loss, and the favorable corner is the design with every factor at its favorable level. A design's service is the completion rate $R_{F_{\min}}$ (written R for short) of a fixed request workload, i.e., the rate at which requests are served with a final fidelity of at least $F_{\min}$. Design optimization then means answering those two questions for each topology over these designs. The answers do not name one design to build, because component costs depend on the vendor and change over time; they show where improvements pay off, and the designer weighs that against current costs.

Q-REFINE also reports the switches and fiber links each topology uses (Table III), so that the designer can price a design.

B. FRAMEWORK OVERVIEW

The answers to the two questions are measured by simulation in the three stages of Fig. 1, namely load selection, a full-factorial DOE that answers the first question, and OFAT sweeps around the favorable corner that answer the second. Load selection fixes the load at which designs are compared, and this load decides how strongly requests compete for the same communication qubits, switches and BSM units.

We set this load with the concurrency limit W, the largest number of requests admitted and not yet ended at any time. Fig. 2 sketches how the completion rate rises, saturates and can collapse as W grows. W is the same for every design of a topology, so that a difference in rate comes from the physical-layer factors and not from a different load, and it is chosen for each topology separately, because the same W can underload one topology and overload another. We choose it at the favorable corner and keep it for the combinations that the DOE and the OFAT sweeps run later, each of which sets every factor at its favorable level or a less favorable one. Requests at the favorable corner need the fewest attempts and release their resources soonest, so the favorable corner needs the most concurrent requests to saturate; at a W that saturates it, the limit holds back no combination before that combination's own resources become the bottleneck. The price is that the less favorable combinations are overloaded, and part of a factor's measured effect then comes from relieving congestion rather than from the better component. We keep that part small by running each topology at W_{sat}, the smallest W whose

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

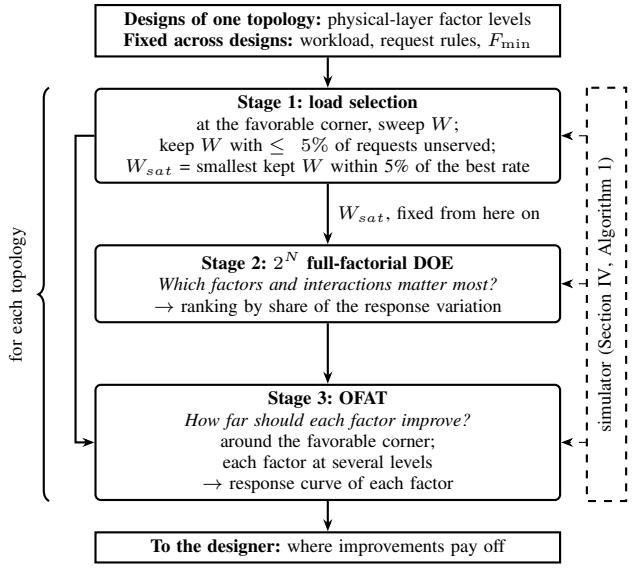


FIGURE 1. The three stages of Q-REFINE, load selection, a full-factorial DOE and OFAT sweeps, run for each topology (Sections III-B and V). W is the concurrency limit, the favorable corner sets every factor at its favorable level, and $N = 11$ is the number of factors. Every stage evaluates its designs with the simulator (dashed arrows).

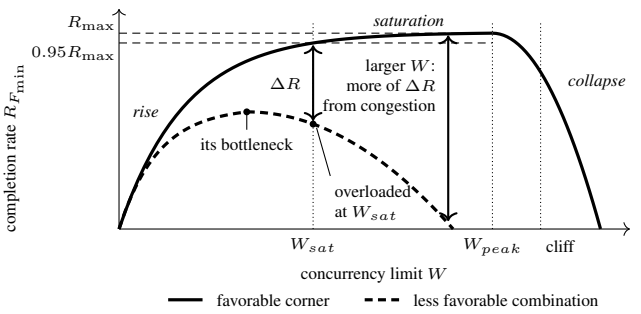


FIGURE 2. Schematic load-response curves, not measured data. As the concurrency limit W grows, the completion rate rises, saturates once shared resources become the bottleneck, and can collapse when waiting requests hold resources while stored pairs decay. W_{sat}, $R_{\max}$, W_{peak} and the cliff are the markers of Sections III-B and V-C. The dashed curve is a less favorable combination, and ΔR marks the difference between the two at a fixed W, which includes rate won back from congestion (Section III-B).

completion rate at the favorable corner is within 5% of its best (Appendix C-A defines these two parts).

Every point of the load-response curve, and every combination of the DOE and the OFAT sweeps, is measured by averaging repeated runs of the simulator with different seeds. A run simulates one batch of the workload on one topology instance, with each physical-layer factor at one level and one value of W; the workload, the request rules (admission, routing, reservation and deadline, Section III-E) and $F_{\min}$ are the same in every run. Within a run, a control plane admits at most W requests at a time and gives each a path and its resources, and a data plane generates the path's links with double-click heralding, stores them and joins them. Each request ends served, unserved when its deadline passes, or infeasible when no path can reach $F_{\min}$. A run returns the completion rate

$$R_{F_{\min}} = \frac{N_{acc}(F_{\min})}{T_{batch}}, \tag{1}$$

where $N_{acc}(F_{\min})$ is the number of requests served with a final fidelity of at least $F_{\min}$ and T_{batch} is the batch duration, from the first request's release to the last request's end. An unserved request adds nothing to N_{acc}, but the time it held resources still counts in T_{batch}, and a run that serves no request has $R_{F_{\min}} = 0$, even when no request can be admitted. A run also returns the unserved and infeasible shares, and load selection uses the unserved share to rule out values of W (Fig. 1).

Three simplifications keep the run focused on the interconnect. First, control decisions take no time, and the only classical signal we time is the herald's return, so requests are served somewhat faster than under a real controller. Second, every QPU converts its photons to the telecom band around 1550 nm, for which standard fiber and commercial optical switches are made, and we count the converter's loss but not its noise (Appendix B-B), so link fidelities are somewhat optimistic. Third, each pair is described by its fidelity alone, the quantity that published application requirements specify. Our storage model slightly overstates this fidelity and our swap model slightly understates it, which changes a multi-segment fidelity by less than 0.01 in either direction (Appendices B-E and B-F). The results therefore describe how the fabric and its physical parameters limit service, not how control protocols, conversion hardware or error structure do.

Sections III-C and III-D define what a run is given, the topology instances and the workload; Sections III-E and III-F define what the control plane and the data plane do. Table II summarizes the notation.

C. TOPOLOGY INSTANCES AND RESOURCE ASSUMPTIONS

We apply the framework to five representative topology families, Fat-tree, Clos, QFly, BCube and DCell (Fig. 3). Following the QDC architecture model of Shapourian et al. [2], each family groups its QPUs into racks, a rack being the set of QPUs under one top-of-rack switch, and the family's other switches and links connect the racks. Each QPU has n_{comm} communication qubits, each with its own fiber to every switch its QPU connects to, and two switches are joined by n_{comm} fibers per link.

In the switch-centric families of Section II-A, Fat-tree, Clos and QFly, a request is generated as one link between its two QPUs. In the server-centric families, BCube and DCell, a path may pass through relay QPUs; it is then split at the relays into segments, each generated as one link between its two end nodes, and each relay joins its two adjacent links by entanglement swapping, a local Bell measurement.

Besides the workload and the factor ranges, which are the same in every run, three items are held fixed across the families, and everything else, such as the number of switches

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

TABLE II. Summary of Notation

Symbol	Description	Symbol	Description
Q_{eff}	Endpoint pool, the QPUs that may be request endpoints.	Δt_{herald}	Time from an attempt's emission until both nodes know its outcome, Eq. (6).
M	Number of two-endpoint inter-QPU requests in a sampled workload batch.	Δt_{cool}	Communication-qubit cooldown time before the local hardware can be used again.
M_{inter}	Number of requests whose endpoints are in different racks.	Δt_{trial}	Attempt interval, Eq. (7).
n_{comm}	Communication qubits per QPU.	T_{gen}	Length of a generation round.
p_{inter}	Target fraction of sampled requests whose endpoints are in different racks.	N_{trial}	Maximum number of attempts in a generation round, Eq. (10).
B	Total of BSM units in a topology, equal to M.	F_{swap}	Fidelity of one entanglement swap on ideal input pairs, Eq. (9).
W	Concurrency limit, the largest number of requests admitted and not yet ended at any time.	$F_{end}(r)$	Final end-to-end fidelity of request r.
$A(t)$	Requests admitted and not yet ended at simulation time t.	$F_{\min}$	Final end-to-end fidelity threshold used to accept a completed request.
T_D	Service deadline, counted from a request's admission (Section III-E).	$\mathrm{Accept}(r)$	Indicator that request r satisfies the final-fidelity threshold.
$\pi,\ \sigma$	A candidate path and one of its segments; $\lvert\pi\rvert$ is its number of segments.	$N_{acc}\,(F_{\min})$	Number of requests served with a final fidelity of at least $F_{\min}$.
$\hat{F}(\pi)$	Best-case fidelity of path π, Eq. (3).	T_{batch}	Batch duration, from the first request's release to the last request's end.
$D_A,\ D_B$	One-way optical delays of the two arms, from each node to the station.	$R_{F_{\min}}$	Completion rate, the response variable, Eq. (1).
L_X	Total loss of arm X, in dB.	$\bar{R}(W)$	Mean completion-rate response at the concurrency limit W over seeded repetitions for one topology.
η_X	Probability that the photon of arm X reaches the station's detectors, Eq. (5).	$\mathcal{W}_{valid}$	Concurrency-limit values at which at most 5% of requests end unserved.
Δt_{cw}	Heralding coincidence window, within which the station counts clicks as one attempt.	$R_{\max}$	Largest valid mean response over the values of W that load selection tries, for a topology.
λ_{dark}	Detector dark-count rate.	W_{sat}	Smallest valid W whose mean response is within 5% of $R_{\max}$; the W of the later stages.
P_{dark}	The probability that a detector produces at least one dark click within the heralding coincidence window of one time bin.	W_{peak}	The W with the largest valid mean response in load selection.
F_0	Fidelity of the emitted qubit–photon pair (Werner model).	$\alpha_f,\ d_{intra},\ d_{inter}$	Fiber attenuation and intra-/inter-rack segment distances.
V_{HOM}	Two-photon (Hong–Ou–Mandel) interference visibility at the station.	$L_{sw},\ L_{conn}$	Switch insertion loss and connector loss.
P_{acc}	Probability that one attempt yields an accepted click record, Section III-F.	τ_{coh}	Memory coherence time; also sets the generation-round length ($T_{gen} = \max(\tau_{coh}, \Delta t_{trial})$) and the storage cut-off.
F_{link}	Fidelity of a heralded link, Section III-F.	$f,\ \eta_{emit},\ \eta_{\mathrm{det}}$	Entanglement-attempt rate, emission efficiency, and detector efficiency.

and links and the structure of the paths, follows each family's construction. The first is the optical band. Every path is in the telecom band (Section III-B), so every path has one set of fiber, switch and connector loss parameters.

The second is the detection hardware. It consists of Bell-state measurement (BSM) units, each a beam splitter with its detectors, attached to an optical switch and reserved by one segment at a time. A segment's station is the switch whose BSM unit measures the segment's two photons. It can be any switch on the segment's path, so the segment's two arms, from each node to the station, can differ in length and loss. Every topology receives the same total of B BSM units, one for every two communication qubits of the request endpoints, so that every link the endpoints can hold at once has a unit; B therefore equals the request count M of Section III-D. The units are split as evenly as possible over the topology's switches, including the small switches that DCell places on its inter-cell links, and the remainder goes to the switches with the most directly attached QPUs. The total never limits requests whose paths have one segment, but a switch's share can.

The third is the pool of request endpoints. Each family's construction admits only particular QPU counts (Table III). We draw request endpoints from 54 QPUs in each, the largest pool all five instances supply; the remaining QPUs are not endpoints, but in the server-centric families they can still serve as relays. Table III lists each instance's construction and hardware.

D. WORKLOAD CONSTRUCTION

The endpoint pool of Section III-C supplies a batch of two-endpoint requests, each asking for an entangled pair between a source and a destination QPU. Two endpoints suffice because gate-model circuits decompose into one- and

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

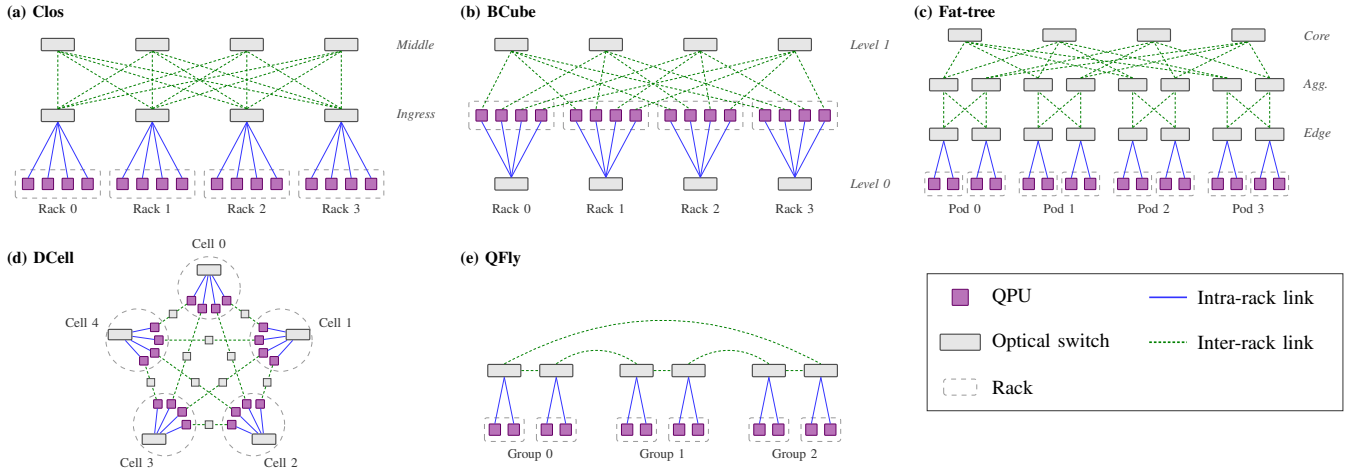


FIGURE 3. Topology-level connectivity abstraction for the five QDC topology families evaluated in this study. The diagrams show QPUs, optical switches, racks, and intra-rack and inter-rack optical links. Every switch operates in the telecom band and hosts Bell-state measurement units, so all switches share one symbol and the labels name their tiers. The top-of-rack switch is an ingress switch in Clos, a level-0 switch in BCube, an edge switch in Fat-tree, the cell switch in DCell and any switch in QFly. In DCell, each inter-cell link passes through a small switch of its own, which hosts that link's BSM units. The instances drawn are small examples of each family; Table III lists the evaluated ones.

TABLE III. Topology Instances, Their Switches and Fiber Links, and the Shared Endpoint Pool

Topology	Construction	QPU-count rule	Total QPUs	Outside pool	Switches	Fiber links (intra / inter)
Fat-tree	k = 6	$k^3/4$	54	0	45	54 / 108
Clos	r = 16, m = 4, n = 4	r·n	64	10	20	64 / 64
QFly	a = 3, p = 3, h = 2	(ah + 1)·ap	63	9	21	63 / 42
BCube	n = 8, ℓ = 1	$n^{\ell+1}$	64	10	16	64 / 64
DCell	n = 7, k = 1	n(n + 1)	56	2	8 + 28	56 / 56

Note: Construction parameters follow the standard definition of each family: Fat-tree radix k; Clos ingress/egress switches r, middle switches m, and QPU ports per ingress switch n; QFly switches per group a, QPUs per switch p, and global links per switch h; BCube and DCell switch port count n, with level ℓ (BCube) or recursion level k (DCell). The QPU-count rule gives the total number of QPUs an instance admits; "Outside pool" is that total minus the shared 54-QPU endpoint pool. A fiber link is an edge of the topology graph. An intra-rack link joins a QPU to its rack's switch, and an inter-rack link is any other link; each link has n_{comm} fibers (Section III-C). DCell has 8 cell switches and 28 small switches, one at the midpoint of each inter-cell link, so each half of that link is $d_{inter}/2$ long. The B BSM units, equal in number across topologies, add two switch ports each and are not counted here.

TABLE IV. Workload settings that are chosen, derived from them, or set by load selection

Quantity	Symbol	Value	Basis
Chosen and held fixed			
Endpoint pool	$\lvert Q_{eff}\rvert$	54	largest pool all five instances supply, i.e., the smallest (Fat-tree, $k = 6$); spread as evenly as possible over racks
Communication qubits per QPU	n_{comm}	5	chosen setting
Inter-rack share	p_{inter}	0.3	chosen setting
Service deadline	T_D	419.6 μs	demonstrated remote-gate timeout [24]; Section V-A
Derived			
Requests	M	135	Eq. (2)
Inter-rack requests	M_{inter}	41	$\lfloor p_{inter} M + \frac{1}{2} \rfloor$
Set by load selection, for each topology			
Concurrency limit	W	W_{sat}	stage 1 of Fig. 1, Eq. (12)

two-qubit gates, so their non-local communication reduces to pairwise requests. Three chosen settings shape a batch of requests, namely the endpoint pool, the number n_{comm} of communication qubits per QPU, which sets the request count, and the share of requests that cross racks.

The workload generator draws request sources and destinations only from the pool Q_{eff} of Section III-C, spread as evenly as possible across the racks so that every rack can supply both intra- and inter-rack requests. Each endpoint appears in as many requests as it has communication qubits, n_{comm}, so that one batch uses every endpoint's communication qubits once and the request count follows from the network rather than from a free choice (Appendix C-B gives the constraints). Hence

$$M = \left\lfloor \frac{\lvert Q_{eff}\rvert \; n_{comm}}{2} \right\rfloor, \tag{2}$$

The value of n_{comm} is a setting we choose, and another value changes M in proportion.

A share p_{inter} of the requests cross racks, and the rest join endpoints in the same rack. No workload trace or quantum-datacenter study we found fixes this share, so it is a setting we choose. Table IV lists these settings, which are held fixed in every experiment.

E. ROUTING, ADMISSION, AND RESOURCE RESERVATION

Before a run, the control plane lists up to two candidate paths for each request, computed once from the topology

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

graph. They are the request's paths of least total optical loss that differ in at least one switch.

Some paths cannot deliver $F_{\min}$ in a design, however quickly their segments are heralded, because every segment, every swap and every wait for a herald costs fidelity. We drop them before the run, using the fidelity a path π would deliver if each of its $|\pi|$ segments σ were heralded after the shortest possible wait and all were joined at once,

$$\hat{F}(\pi) = F_{swap}^{|\pi|-1} \prod_{\sigma\in\pi} \left[\tfrac{1}{2} + \left(F_{link}(\sigma) - \tfrac{1}{2}\right) e^{-\Delta t_{herald}(\sigma)/\tau_{coh}}\right], \tag{3}$$

where F_{link}, Δt_{herald}, τ_{coh} and F_{swap} are the link fidelity, herald time, decay time and swap fidelity of Section III-F, and each segment's factor is taken at the station on its path that maximizes it. A longer wait only moves a fidelity further toward $1/2$, and $F_{\min} > 1/2$, so waiting cannot lift such a path to $F_{\min}$; on a one-segment path, $\hat{F}$ is the fidelity the path delivers when its segment uses that station. A request with no path left is infeasible and is never admitted, so "infeasible" means unreachable at the stated parameters, while a reachable but slow request shows as unserved.

During the run, the control plane admits pending requests in the order of the seeded batch, so that simultaneous requests are resolved reproducibly rather than by an optimization policy. Let $A(t)$ be the set of requests admitted and not yet ended at simulation time t. Admission keeps

$$|A(t)| \le W, \tag{4}$$

so a pending request is admitted as soon as an admitted one ends.

An admitted request tries its candidate paths in order of total loss. The first reservation of a path is all or none, so that no request starts generating while it waits for resources that another request holds. A path needs a communication qubit at each end of each segment, which includes two qubits at every relay QPU, a BSM unit at each segment's station, and a port on every switch that each photon crosses. Each segment's station is chosen at reservation among the switches on its path that have a free BSM unit, preferring the one whose longer arm is shortest. If no candidate path can be reserved, the request waits and tries again when resources are released. Once a segment's link is heralded, its station and ports are released and the link waits in its two communication qubits; if a stored link expires, its segment reserves a station and ports again and may wait for them. The control plane runs no congestion-control protocol and gives no topology a traffic policy of its own, so congestion arises only from contention for these resources, and the load at which a topology saturates reflects its structure and physical parameters.

The limit W bounds how many requests are in service, but not how long each one stays. A computation cannot wait indefinitely for a remote gate, because its data qubits decohere while the link is generated, so an admitted request that is not served within the service deadline T_D of its admission ends unserved, releases all its resources and makes room for the next pending request. T_D is a property of the workload, the time the data qubits can wait, and not of the design, so the memory coherence time τ_{coh} changes how fast stored links decay but not how long a request may wait. Section V-A gives its value.

F. ENTANGLEMENT GENERATION, HERALDING, AND FIDELITY MODEL

Once a path is reserved, the data plane generates a link on each of its segments.

Every link is generated by double-click (Barrett–Kok) heralding [25]. Both communication qubits emit a photon toward the segment's station, and clicks at the station's detectors herald the link. In the naming of Jones et al. [26], this is a meet-in-the-middle link, as opposed to a sender-receiver or a midpoint-source link; QuISP calls it a memory–interface–memory (MIM) link (Table I). We prefer it to single-click heralding because a lost photon lowers only its rate, not its fidelity, and its optical phase need not be stabilized along each switched path (Appendix B-A).

Each photon travels only from its node to the station, so it suffers only the loss of its own arm. We collect every loss between the emitter and the node's output fiber into one parameter, the emission efficiency η_{emit}, which includes emission into the collected mode, collection, coupling into fiber and frequency conversion to the telecom band. For arm $X \in \{A, B\}$, the simulator adds the fiber attenuation, switch insertion loss and connector loss along the arm into L_X (in dB), including the station switch, which each photon crosses to reach the BSM unit. The photon then reaches the station's detectors with probability

$$\eta_X = \eta_{emit}\, 10^{-L_X/10}. \tag{5}$$

The station has two detectors, each read in an early and a late time bin, and it accepts the four of the 2^4 click records in which exactly one detector clicks in each bin. Each accepted record heralds a known Bell state, Ψ^+ when the same detector clicks in both bins and Ψ^- otherwise. A heralding event is triggered when the station observes an accepted record. Dark clicks can also produce such a record, and the record cannot tell them from photon clicks, so every heralded link receives one average fidelity, F_{link}, which false heralds lower. For a detector with a dark-count rate λ_{dark}, the probability of at least one dark click within the heralding coincidence window Δt_{cw}, the window in which the station reads each time bin, is $P_{dark} = 1 - \exp(-\lambda_{dark}\, \Delta t_{cw})$.

The link model takes the fidelity F_0 of the emitted qubit–photon pair, the two-photon Hong–Ou–Mandel (HOM) interference visibility V_{HOM}, the two arm transmissions η_A and η_B, the detector efficiency η_{det}, and P_{dark}. We model the source as a Werner channel; at the same F_0 it gives a lower F_{link} than pure dephasing and nearly the lowest of any Pauli error (Appendix B-C). From these parameters we compute in closed form (Appendix B-C) the acceptance probability $P_{acc}(\sigma)$ that one attempt on segment σ yields

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

an accepted record, and its link fidelity $F_{link}(\sigma)$. Without dark counts, $P_{acc} = \frac{1}{2}\,\eta_A\eta_B\eta_{det}^2$ and

$$F_{link} = \frac{G\,(2V_{HOM}+1) + V_{HOM} + 5}{18},$$

with $G = 4F_0(2F_0-1)$. This F_{link} does not depend on loss, so loss costs rate but not fidelity, and $F_{link} = 1$ when $F_0 = V_{HOM} = 1$; when both photons are lost, every accepted record is a coincidence of dark clicks and $F_{link} = 1/4$.

The probability model determines whether a heralding event occurs, while the timing model determines how long the reserved resources remain occupied before the outcome is known. Let D_A and D_B denote the one-way optical delays of the two arms, from each node to the station. The nearer node delays its emission so that both photons reach the station together, and both nodes know the outcome after

$$\Delta t_{herald} = 2\max(D_A, D_B) + \Delta t_{cw}. \tag{6}$$

A subsequent attempt cannot begin until this outcome is known and the local hardware is ready for use again. Let Δt_{cool} denote the communication-qubit cooldown time, the time to re-initialize a communication qubit, and let f denote the entanglement-attempt rate. A communication qubit is re-initialized only after the herald reports a failed attempt, and it cannot emit faster than f, so the attempt interval is

$$\Delta t_{trial} = \max\Big(\Delta t_{herald} + \Delta t_{cool},\ \tfrac{1}{f}\Big), \tag{7}$$

and an attempt with a click also waits for the station's detectors to recover, which lengthens it only when Δt_{trial} is shorter than the recovery time (Appendix B-D).

A heralded link is stored in its two communication qubits and decays by pure dephasing, the phase-damping part of the memory model of Avis et al. [22] (Appendix B-E), so that its fidelity at time t is

$$F(t) = \tfrac{1}{2} + \big(F_{link}(\sigma) - \tfrac{1}{2}\big)\, e^{-(t-t_e)/\tau_{coh}}, \tag{8}$$

where t_e is the emission time of the successful attempt and τ_{coh} the decay time of the pair's fidelity. A link still stored at age τ_{coh} is discarded and its segment regenerated. Its fidelity is then at most $\frac{1}{2} + \frac{1}{2e} \approx 0.68$, below $F_{\min}$, so the cut-off discards no link that could still reach $F_{\min}$.

The swap succeeds deterministically, because the relay performs it as a local Bell measurement with gates and measurements on its own qubits [22]. The joined link has fidelity

$$F_{out} = F_L(t)\, F_R(t)\, F_{swap}, \tag{9}$$

where $F_L(t)$ and $F_R(t)$ are the two links' fidelities at the swap time, Eq. (8), and F_{swap} is the fidelity of the swap itself on two ideal input pairs; SeQUeNCe uses the same product form [15]. The joined link then decays from the swap time.

The final end-to-end fidelity $F_{end}(r)$ of request r is the fidelity of the link that spans the request, at the moment that link is formed. On a one-segment path this is the heralded link; otherwise it is the link that Eq. (9) leaves once all segments are joined. Request r is accepted, written $\mathrm{Accept}(r)$, when $F_{end}(r) \geq F_{\min}$. A request that is not accepted releases its link and its reservation. It then tries its candidate paths again, until it is served or its service deadline passes.

IV. IMPLEMENTATION

This section describes how the models of Section III are assembled into one simulator. Three demands shape it. First, every topology must be run under the same request handling, so that its results reflect its fabric and physics rather than code written for it. Second, a pair must cost far fewer events than attempts (Section II-C). Third, the three stages of Fig. 1 repeat a run for every concurrency limit, design and replication, so every run must be independent and reproducible.

A. SIMULATOR ARCHITECTURE

The Q-REFINE simulator is a discrete-event simulator written in Python, built, following Avis et al. [22], from modules that act as services, each with a defined input and output (Appendix D). The request controller, routing and resource modules form the control plane of Section III-B, and link generation, storage and swapping the data plane; a workload builder, an event core, a metrics recorder and an experiment driver complete it. A design enters only through the modules that build the topology and its resources, route requests and generate, store and join links. The request controller, which admits requests and keeps the only per-request state, is the same for every design, and a switch-centric path is a path of one segment, so one controller serves all five topologies. This meets the first demand.

B. ONE SIMULATION RUN

Algorithm 1 gives one run in the order of the models of Section III and cites each model's equation rather than restating it. Besides the symbols of Table II, it uses G for the topology instance and $\mathcal{R}$ for its qubits, BSM units and ports, θ for the design's factor levels, $\mathcal{B}$ for the request batch, Π_r for the candidate paths of request r, X for the attempts to the first herald, t_X for the emission time of attempt X and t for the simulation time.

Link generation (lines 17–20) meets the second demand. Generation runs in rounds of length T_{gen}, and a round holds the attempts that start within T_{gen}, at most

$$N_{trial} = \max\left(1,\ \left\lfloor \frac{T_{gen}}{\Delta t_{trial}} \right\rfloor\right) \tag{10}$$

attempts. For each round, the simulator draws X from a geometric distribution with parameter $P_{acc}(\sigma)$ and schedules one event, the herald if $X \leq N_{trial}$ and otherwise the next round, which starts at the end of the round's last attempt on the resources already held. A segment thus costs one event per round rather than one per attempt, and because a request ends at its deadline, at most about T_D/T_{gen} rounds however small P_{acc} is. Attempts are independent in our model, because each starts from re-initialized communication qubits,

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

Algorithm 1 One Q-REFINE run

Require: G; θ; Q_{eff}, n_{comm}, p_{inter}; W; $F_{\min}$; T_D; a seed
Topology and resources (III-C); workload (III-D)
$\mathcal{R} \leftarrow$ qubits (n_{comm} per QPU), BSM units, ports of G
$\mathcal{B} \leftarrow M$ requests by Eqs. (C.2)–(C.5)
Routing, admission and reservation (III-E)
for each request $r \in \mathcal{B}$ **do**
  $\Pi_r \leftarrow$ candidate paths π with $\hat{F}(\pi) \geq F_{\min}$ ▷ (3)
  if $\Pi_r = \emptyset$ **then** mark r INFEASIBLE; it is never admitted
procedure ADMIT ▷ admission event
  while $|A(t)| < W$, Eq. (4), and a request is pending **do**
    move the next pending r into $A(t)$
    DEADLINE(r) at $t + T_D$; DISPATCH(r)
procedure DISPATCH(r)
  if some $\pi \in \Pi_r$, tried in order of total loss, has every qubit, BSM unit and port it needs free
    then reserve them; GENERATE(σ) for each σ of π
    else DISPATCH(r) at the next release
procedure END(r, state) ▷ SERVED or UNSERVED
  release all that r holds; remove r from $A(t)$; ADMIT
on DEADLINE(r): **if** $r \in A(t)$ **then** END(r, UNSERVED)
Generation, heralding and fidelity (III-F)
procedure GENERATE(σ) ▷ Eqs. (7), (10)
  $X \sim$ Geometric($P_{acc}(\sigma)$) ▷ (B.1)
  if $X \leq N_{trial}$ **then** HERALD(σ) at $t_X + \Delta t_{herald}$
  else GENERATE(σ) after the round's last attempt
on HERALD(σ) for request r:
  release σ's station and ports; store the link in its qubits
  its fidelity is $F_{link}(\sigma)$ at emission, then decays ▷ (B.2), (8)
  at age τ_{coh} from emission: discard it; then
    re-reserve σ's station and ports (may wait); GENERATE(σ)
  join adjacent stored links with F_{swap} ▷ (9)
  if one link spans r: **if** Accept(r) **then** END(r, SERVED)
    else release the link and the reservation; DISPATCH(r)
Event loop and response (III-B)
ADMIT; handle events in time order until all requests end
return $R_{F_{\min}}$, Eq. (1); unserved and infeasible shares

so the draw is exact and the time of the first herald does not depend on T_{gen}.

C. REPRODUCIBILITY

The experiment driver runs the three stages of Fig. 1 on top of Algorithm 1 and meets the third demand. Event times are integers in picoseconds, and simultaneous events are handled in the order they were scheduled, so a run is a deterministic function of its design, concurrency limit and seed. Runs share no state, so they execute in parallel. The batch of replication i is drawn from a seed that depends only on i, so all designs of a topology face the same requests (common random numbers), and a difference between designs does not come from the draw of requests.

V. EXPERIMENTAL DESIGN

This section fixes the choices that Section III leaves to the study. Section V-A gives the settings every run shares, Section V-B the factors and the fidelity threshold, Section V-C the rules of load selection, and Section V-D those of the DOE and the OFAT sweeps.

TABLE V. Common Evaluation Settings

Item	Configuration
Topologies and workload	Tables III and IV
Candidate paths	Up to two per request, computed once before each run
Concurrency limit	W is swept during load selection; DOE/OFAT use topology-specific W_{sat}
W sweep grid	W in {1, 2, 4, 7, 10, 14, 20, 27, 41, 54, 68, 81, 108, 135}
BSM units per topology	$B = 135$, one per two endpoint communication qubits; split evenly over the switches (Section III-C)
Fidelity threshold	Final end-to-end request fidelity must be at least 0.85 [27]
Fixed physical settings	Detector efficiency 0.98, detector dead time 58 ns, heralding coincidence window 15 ns, fiber attenuation 0.18 dB/km, intra-rack distance 2 m
Generation round	$T_{gen} = \max(\tau_{coh}, \Delta t_{trial})$, so a round holds at least one attempt, Eq. (10)
Heralding	Time-bin double-click at the station of each segment
Repetitions	30 runs with different seeds per load and per DOE combination; 200 per OFAT point

A. MAIN EVALUATION SETTING

Table V lists the settings that every run shares. The topology instances and the workload are those of Tables III and IV, and the final-fidelity threshold $F_{\min} = 0.85$ is justified in Section V-B. Of these settings, the service deadline T_D of Section III-E needs its source and a robustness check, which we give here.

We set $T_D = 419.6$ μs, the timeout of a demonstrated remote gate between qubit registers [24]. After 50 entanglement attempts of 8.392 μs, that experiment abandons generation and re-initializes its data qubits; its authors chose this value to balance gate rate against fidelity. The results thus describe computations whose data qubits can wait a few tenths of a millisecond for a remote gate. No source fixes one deadline for every application, so this value is a chosen scenario, not a universal constant, and we repeat load selection at two other values, also varying the 5% limit on unserved requests to 2% and 10% (Appendix C-C). One is 259 μs, the time to generate one entangled pair assumed by a distributed-computing compiler [28]; the other is 2 ms, the error-correction cycle targeted for modular neutral-atom processors [29].

B. PHYSICAL FACTORS AND FIDELITY THRESHOLD

The sensitivity analysis varies the eleven physical-layer factors of Table VI, which fall into three groups by how they act on a request. Four set the fidelity of the pair a request receives, namely the emitted-pair fidelity F_0, the HOM visibility V_{HOM}, the memory coherence time τ_{coh} and the swap fidelity F_{swap}. Three set how often a link is heralded. The emission efficiency η_{emit} sets an attempt's chance of success, and the attempt rate f and the communication-qubit cooldown time Δt_{cool} set, together with Δt_{herald}, the

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

TABLE VI. Physical-Layer Factors and the Two Levels Used in the DOE

Factor	Symbol	Less favorable	Favorable	Unit
Emitted-pair fidelity	F_0	0.94	0.99	–
HOM visibility	V_{HOM}	0.89	0.955	–
Memory coherence time	τ_{coh}	3×10^{-7}	75	s
Swap fidelity	F_{swap}	0.95	0.97	–
Emission efficiency	η_{emit}	0.0505	0.163	–
Attempt rate	f	0.161	10.9	MHz
Communication-qubit cooldown time	Δt_{cool}	5.8	0.83	μs
Switch insertion loss	L_{sw}	1.5	0.35	dB
Connector loss	L_{conn}	0.75	0.2	dB
Inter-rack distance	d_{inter}	100	3	m
Dark-count rate	λ_{dark}	10^3	10^{-4}	Hz

Note: The favorable level of each factor is the favorable end of its reported range (for τ_{coh} the runs used 75 s, above the reported 63 s), and together these levels form the favorable corner. The less favorable level is the other end of the range, except for η_{emit}, f and F_0 (Section V-D). Appendix F gives each factor's reported range and sources, how the less favorable level is chosen from it, and the range of each OFAT sweep.

attempt interval, Eq. (7). The other four act on a segment's arms and its station, namely the switch insertion loss L_{sw}, the connector loss L_{conn}, the inter-rack distance d_{inter} and the dark-count rate λ_{dark}; Section III-F defines all eleven. Table VI lists the two levels at which the DOE runs each factor. The favorable level is the best value reported for demonstrated or projected hardware, and Appendix F gives each factor's reported range, its sources and how the levels are chosen from it. Five other quantities are fixed at the values of Table V. The detector efficiency has a reported range too narrow to change the rate (0.98–0.99), and the detector dead time of 58 ns is shorter than any cooldown time, so it never delays an attempt. The fiber attenuation adds at most 0.02 dB over 100 m, the intra-rack distance of 2 m adds a negligible delay, and the coincidence window of 15 ns is that of a demonstrated experiment.

A request counts toward $R_{F_{\min}}$, Eq. (1), only if the pair it delivers is usable, and the fidelity that makes a pair usable depends on its use (Appendix C-D). We take the requirement of the application a QDC is built for, modular fault-tolerant computation. Surface-code patches joined by undistilled Bell pairs reach threshold with about 10% bit- and phase-flip probability at the interface [27]; each flip of a Werner pair occurs with probability $2(1-F)/3$, so 10% corresponds to $F_{\min} = 0.85$. Because our simulator does not model distillation, the delivered pair itself must reach $F_{\min}$, and because at threshold a larger code no longer suppresses errors, the completion rate counts requests that meet a necessary condition for fault-tolerant use, not a comfortable operating point.

C. LOAD SELECTION

In load selection, every factor is held at its favorable level, the favorable corner of Section III-A, while W is varied over the grid listed in Table V. For each topology and value of W, we average 30 runs with different seeds. With 30 runs, the standard error of the mean is 0.7–1.7% of $R_{\max}$, well below the 5% margin in Eq. (12). The response at each W is the completion rate of Eq. (1).

For each topology, let $\bar{R}(W)$ denote the mean response at the concurrency limit W over the seeded repetitions. We include a W value in $\mathcal{W}_{valid}$ when at most 5% of its requests end unserved; the infeasible share does not depend on W, because pruning fixes it before the run, and is reported separately. Over this valid set, we define

$$R_{\max} = \max_{W \in \mathcal{W}_{valid}} \bar{R}(W). \tag{11}$$

The selected concurrency value is defined as

$$W_{sat} = \min\{\, W \in \mathcal{W}_{valid} : \bar{R}(W) \geq 0.95\, R_{\max} \,\}. \tag{12}$$

We also record diagnostic markers for the peak response and possible overload behavior. The peak-response concurrency is

$$W_{peak} = \arg \max_{W \in \mathcal{W}_{valid}} \bar{R}(W). \tag{13}$$

A cliff point is recorded when a value of W larger than W_{peak} either fails the validity condition or falls below 80% of $R_{\max}$, i.e., the smallest W beyond the peak at which the load-response curve collapses rather than degrading gracefully, as sketched in Fig. 2. The cliff marks the upper edge of the useful range of W. When W_{sat} lies at the edge of the tested grid, it is the largest tested value that meets the rule rather than a confirmed saturation point.

D. SENSITIVITY METHODOLOGY

1) Full-Factorial DOE for Factor Attribution

First, we run a full-factorial design of experiments across the eleven physical-layer factors. Each factor is evaluated at its favorable level and its less favorable level (Table VI), giving 2^{11} or 2048 physical-factor combinations per topology. For eight factors the less favorable level is the less favorable end of the reported range. For η_{emit}, f and F_0 that end would leave most combinations with no request served and nothing to attribute, so their level is the least favorable value at which every topology, with the other factors at their favorable levels, keeps at least 10% of its favorable-corner rate, found by a preliminary sweep before the DOE. Because the rate jumps with F_0 rather than declining, this value is 0.94, the first grid value above 0.931, where a one-segment path first reaches $F_{\min}$ (Appendix F). For each combination, the simulator records the completion rate of Eq. (1). Each combination is run 30 times with different seeds. With N factors, each main effect compares the mean response of the 2^{N-1} combinations at a factor's favorable level with that of the 2^{N-1} at its less favorable level, so each side averages $2^{N-1} \times 30$ runs, and each interaction is estimated from halves of the same size. Thirty runs per combination therefore already estimate the effects tightly, and more would multiply the cost of all 2^N combinations.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

The rate cannot be read as a sum of effects over all combinations, because some combinations serve no request. When the factors that set pair fidelity are too low, no path reaches $F_{\min}$ and $R = 0$ whatever the other factors are, so in those combinations the other factors appear to interact with them, although they only lose their chance to act. We therefore analyze the DOE in two steps. The first finds which combinations serve any request. The second takes the 2^{10} combinations with F_0 at its favorable level, which all serve requests (Section VI-B), so $\ln R$ is defined, and fits a multiplicative model, i.e., additive effects on $\ln R$ [20]. The two steps form a hurdle model, a rule for whether the rate is zero followed by a model of its size where it is not. Each main effect then reads as a gain, the ratio by which R grows when one factor moves from its less favorable level to its favorable level, averaged over the other factors' levels, and an interaction reads as a gain that depends on another factor's level. We read every effect in this direction of improvement; for the cooldown time, the losses, the distance and the dark-count rate, improving means decreasing the value. Appendix G gives the allocation of variation over all 2047 effects of the full design. Two levels show the response only at those levels; the OFAT sweeps below trace it along each range.

2) OFAT Sweeps for Response Curves

Then, we run one-factor-at-a-time (OFAT) sweeps around the favorable corner to observe how the response changes across each physical factor's range. In each OFAT sweep, one physical-layer factor is varied over its reported range up to its favorable end, on a logarithmic grid where the range spans orders of magnitude; for η_{emit}, f and F_0 the sweep starts one grid step below the level at which the rate falls under 1% of the favorable-corner rate (Appendix F), so each curve covers the whole region in which the factor limits the rate. The remaining factors are kept at their favorable levels, the point at which load selection chooses W. Because every other factor is at its best level, a curve shows how far the swept factor alone limits the rate.

Each point of an OFAT curve rests on its own runs alone, so it needs more of them than a DOE combination. We use 200 runs per point, which narrow its confidence interval by a factor of $\sqrt{200/30} \approx 2.6$ relative to 30. The DOE answers the first design question at the two DOE levels, and the OFAT curves answer the second by showing where along each range the gain is earned.

VI. RESULTS

Section VI-A gives each topology's load, and Sections VI-B and VI-C answer the two design questions of Section III-A for each topology at that load.

A. SELECTED LOADS

Table VII gives each topology's load selection at the favorable corner. For Clos, QFly and BCube every tested W is valid, i.e., leaves at most 5% of requests unserved, and W_{sat} marks where the load-response curve flattens. For Fat-tree and DCell the unserved share passes 5% at $W = 81$ and $W = 54$, so $\mathcal{W}_{valid}$ ends at 68 and 41 while the rate still grows; $R_{\max}$, and with it W_{sat}, is then set by the 5% limit on unserved requests rather than by saturation. At W_{sat}, 87–100% of batches contain a request that misses its deadline, and a batch lasts 1.01–1.36 times T_D, so the batch duration stays near T_D and R changes almost only with the number of requests served in time. A factor therefore raises R mainly by letting more requests finish in time, and Table VII gives the unserved and infeasible shares at W_{sat}. DCell's infeasible requests, 26.8% of its workload, are those whose paths have three segments, which by Eq. (3) deliver at most 0.828 even at the favorable corner, below $F_{\min}$; Appendix G gives the number of segments of each topology's paths and the resources its requests wait for. These loads hold for $T_D = 419.6$ μs; at 259 μs no W keeps Fat-tree, Clos, QFly or BCube within the 5% limit (Appendix C-C).

TABLE VII. Selected Loads At The Favorable Corner

Topology	W_{sat}	W_{peak}	$R(W_{sat})$ (k/s)	$R_{\max}$ (k/s)	Cliff W	Unserved (%)	Infeasible (%)
Fat-tree	54	68	244.3	256.7	81	3.4	0
Clos	81	135	293.8	307.4	–	2.1	0
QFly	108	135	313.3	315.8	–	2.0	0
BCube	81	135	292.8	307.7	–	1.8	0
DCell	41	41	166.6	166.6	54	3.2	26.8

Note: Rates are thousands of completed requests per second, mean over 30 seeds, and shares are of all requests at W_{sat}; Section V-C defines W_{sat}, W_{peak}, $R_{\max}$ and the cliff, and a dash means no cliff within the tested W. Each topology runs at its own W_{sat}, so rates are not comparable across rows.

B. WHICH FACTORS CHANGE THE RATE

The first step of the analysis finds which of the 2048 combinations serve any request. In every topology 768 serve none, exactly those with $F_0 = 0.94$ and at least one of V_{HOM} and τ_{coh} at its less favorable level, so these three factors decide together whether any path can reach $F_{\min}$. At $F_0 = 0.94$ a one-segment path reaches $F_{\min}$ only if V_{HOM} is high and the pair loses little fidelity during Δt_{herald}, so lowering either one ends all service; at $F_0 = 0.99$ the link keeps enough margin for both. The margin a path needs grows with its number of segments, because every swap compounds the fidelity losses of the links it joins.

In the 1024 combinations with $F_0 = 0.99$ every combination serves requests, and $\ln R$ is close to a sum of single-factor effects. Five main effects and the interaction of the attempt rate and the cooldown time explain 96–100% of the variation of $\ln R$ in every topology (Table VIII), so each factor multiplies the rate by a gain of its own. The emission efficiency has by far the largest gain, 8.4–9.5, followed by the cooldown time and the attempt rate at 2.1–2.6 each and the switch insertion and connector losses at 1.7 and 1.3. Across these combinations $\ln R$ rises almost linearly with the logarithm of the rate of successful heralds on the

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

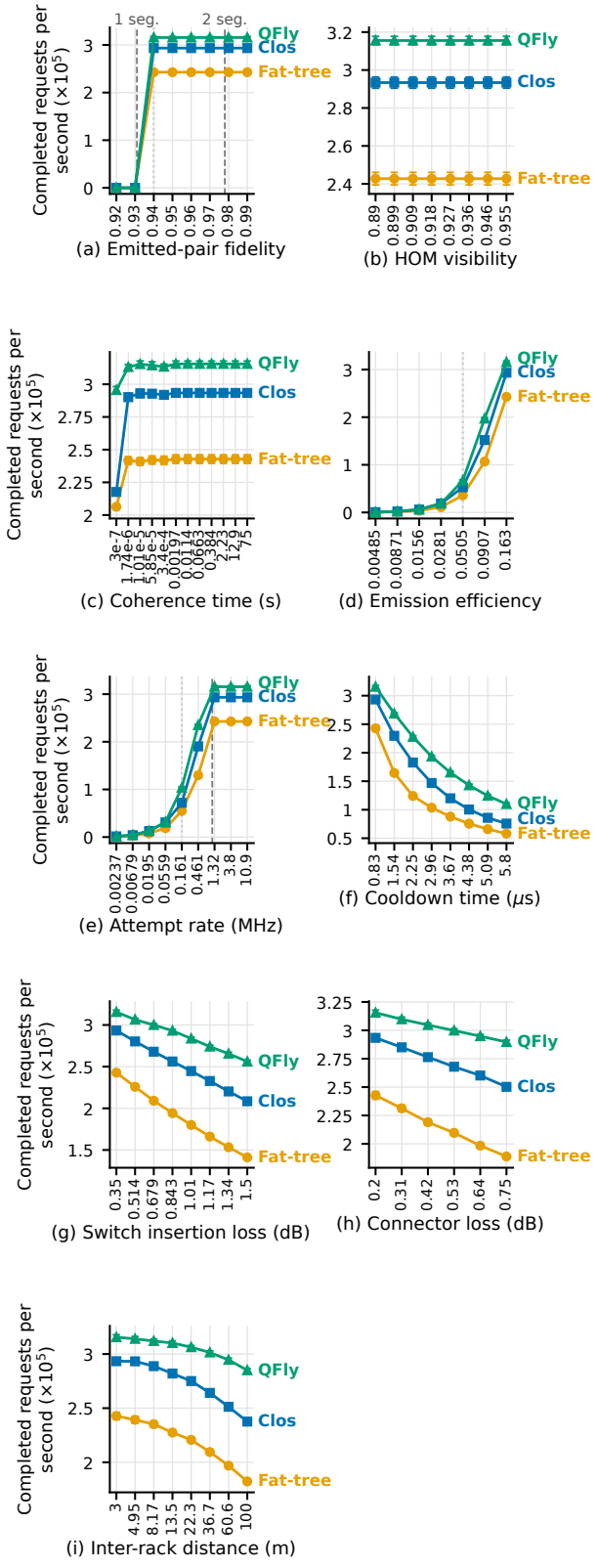


FIGURE 4. OFAT response of Fat-tree, Clos and QFly, each at its W_{sat}, over each factor's sweep range (Appendix F), with the other ten factors at their favorable levels; each point is the mean over 200 seeds. Dashed lines mark the F_0 at which one- and two-segment paths first reach $F_{\min}$ (0.931 and 0.978) and the attempt rate $1/(\Delta t_{herald} + \Delta t_{cool})$; dotted lines mark the less favorable level of the DOE. Each topology runs at its own W_{sat}, so rates are not comparable across topologies. F_{swap} and λ_{dark} leave the rate unchanged and are not shown.

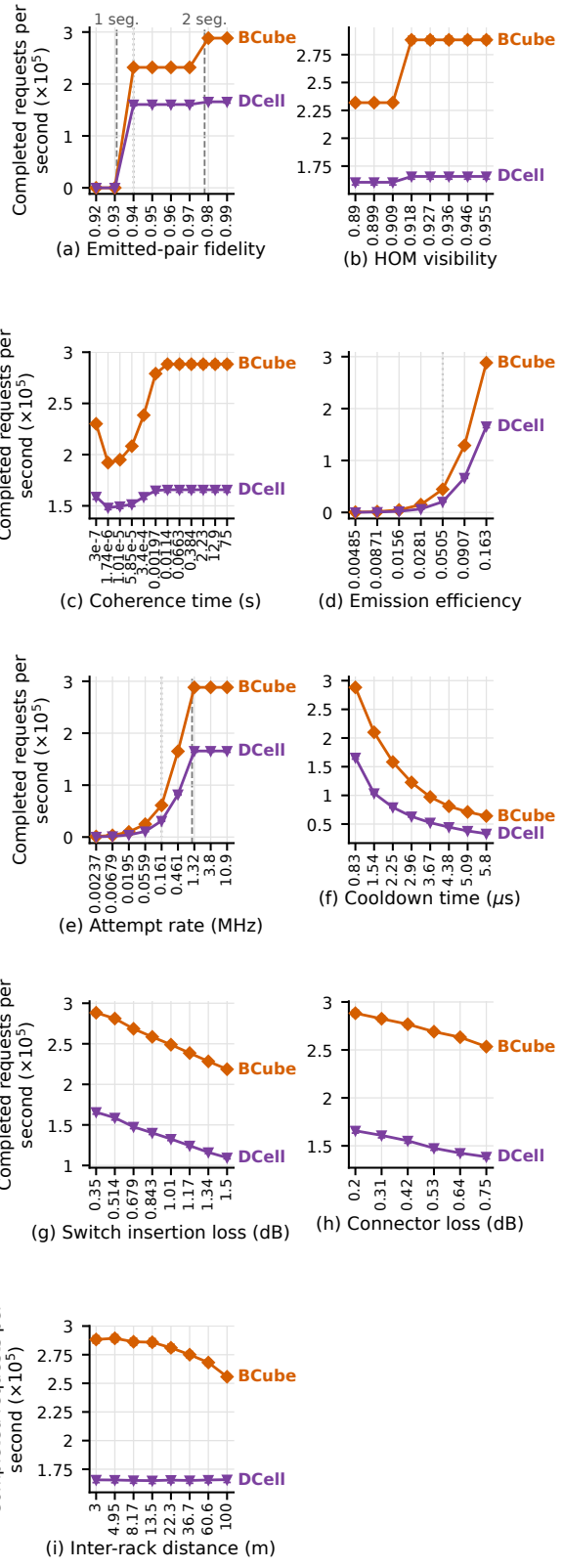


FIGURE 5. OFAT response of BCube and DCell, each at its W_{sat}, over each factor's sweep range, with the other ten factors at their favorable levels; each point is the mean over 200 seeds, and lines are marked as in Fig. 4. BCube's steps at $F_0 = 0.978$ and $V_{HOM} = 0.918$ are where its two-segment requests first reach $F_{\min}$. The dip of the coherence-time curves near 1 μs is explained in Section VI-C. Each topology runs at its own W_{sat}, so rates are not comparable across topologies.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

served paths, $P_{acc}/\Delta t_{trial}$, with slope 0.88–0.95 and R^2 of 0.96–1.00, so these factors act on R through that rate. V_{HOM}, τ_{coh}, F_{swap}, d_{inter} and λ_{dark} each change R by less than 25% when $F_0 = 0.99$; the few gains below 1, such as that of τ_{coh} for QFly, arise because a very short coherence time makes requests infeasible before they take resources, which leaves more capacity for the others. These gains compare the two levels of Table VI. For η_{emit} and f the less favorable level keeps at least 10% of the favorable-corner rate (Section V-D), which bounds their gains at the favorable corner near tenfold, so the gains rank the factors over these spans rather than over their full ranges.

The interaction of the attempt rate and the cooldown time follows from Eq. (7). An attempt starts only after both nodes know the outcome of the previous one and the communication qubit has been re-initialized, so the attempt interval is the larger of $1/f$ and $\Delta t_{herald}+\Delta t_{cool}$. A higher attempt rate pays off only until $1/f$ falls below that sum, and a shorter cooldown time only when the attempt rate is high enough to use it; the gain of f is 4.7–6.2 with the cooldown time at 0.83 μs but only 1.05–1.06 at 5.8 μs. The two must therefore be improved together.

C. OVER WHAT RANGE IMPROVEMENTS PAY OFF

The DOE gains compare the response only at the two DOE levels; Figs. 4 and 5 show where along each range a factor stops paying, with every other factor at its favorable level. F_0 and V_{HOM} act as steps. The rate is zero at $F_0 \leq 0.93$ and jumps at 0.94, where one-segment paths first reach $F_{\min}$; with the other factors at their favorable levels, the lowest F_0 at which a path reaches $F_{\min}$, Eq. (3), is 0.931 for one segment and 0.978 for two. BCube regains 80% of its rate at 0.94 and the rest at 0.98, where its two-segment requests, about a quarter of its workload, can be served, and its V_{HOM} curve steps at 0.918 for the same reason; above its step neither factor raises the rate. Over its range F_{swap} moves no request across $F_{\min}$ and leaves the rate unchanged. The coherence time matters over a wider range for BCube and DCell, whose paths with more than one segment store one link while the next is generated; their rate keeps rising up to about 10 ms, whereas the rate of Fat-tree, Clos and QFly, whose paths have one segment, is flat above 1.7 μs and lower only at 0.3 μs. As τ_{coh} falls toward about 1 μs the rate of BCube and DCell falls, and at 0.3 μs it partly recovers, because requests whose stored links would decay below $F_{\min}$ then become infeasible before they take any resource, whereas at longer coherence times they are admitted, hold resources and then fail (for BCube, 24% of requests are infeasible and 2% unserved at 0.3 μs, against none infeasible and 27% unserved at 1.7 μs).

Within its range the emission efficiency does not saturate, so raising it beyond 0.163 would still raise the rate. The rate stops rising with f between the grid points 0.46 and 1.3 MHz, consistent with $1/(\Delta t_{herald}+\Delta t_{cool}) \approx 1.1$ MHz at the favorable corner, and beyond that only a shorter cooldown time raises it. Over its range the cooldown time alone leaves between 20% and 35% of the favorable-corner rate.

At its less favorable end the switch insertion loss leaves 58–81% of the rate and the connector loss 78–92%, and neither curve flattens, so improving either pays off over its whole range. At 100 m the inter-rack distance leaves 75–100% of the rate; the longer time until both nodes learn each attempt's outcome predicts part of this drop, and the rest comes with more requests missing the deadline. The dark-count rate leaves the rate unchanged, because at 1 kHz a 15 ns window holds a dark count with probability 1.5×10^{-5}, far below the probability of a real herald.

VII. DISCUSSION

The first implication is that load selection must precede the attribution of physical factors. A single concurrency limit for all topologies would mix load with physical-layer sensitivity (Section III-B). The selected load also depends on the service deadline (Appendix C-C), so a study must state the deadline its applications allow before it evaluates designs.

For design, the three groups of factors of Section V-B play different parts. Of the four that set pair fidelity, F_0, V_{HOM} and τ_{coh} decide together whether any request is served, and F_{swap} leaves the rate unchanged over its range. F_0 and V_{HOM} pay off up to their step, which the longest paths that can reach $F_{\min}$ set, and τ_{coh} raises the rate of paths with more than one segment up to about 10 ms. The three that set how often a link is heralded multiply the rate. The emission efficiency gives the largest gain and was still rising at the best reported value, and the attempt rate pays off only with a short cooldown time. Of the four that act on the arms and station, the switch insertion and connector losses and the inter-rack distance cost rate, and the dark-count rate does not.

The topology enters these answers through its paths rather than through a different set of important factors. The emission efficiency, the attempt rate and the cooldown time lead in all five topologies; what differs is the number of segments in a path. At the scale evaluated every path of Fat-tree, Clos and QFly has one segment, so the swap fidelity cannot matter and, with the other factors at their favorable levels, $F_0 \geq 0.931$ suffices; BCube's two-segment paths need $F_0 \geq 0.978$, and DCell's three-segment paths cannot reach $F_{\min}$ at any tested level, which makes 26.8% of its workload infeasible.

The study should be interpreted within its modeling scope. The workload abstracts inter-QPU communication demand as two-endpoint, non-local requests, rather than simulating the full application execution time using local gates. This choice makes workload pressure controllable and supports sensitivity analysis, but it does not replace circuit-level validation. The simulator also gives control decisions no time and omits conversion noise and the error structure of a pair (Section III-B). The results also rest on inter-rack distances of at most 100 m, a 15 ns coincidence window and

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

TABLE VIII. 2^{10} Full-Factorial DOE With $F_0 = 0.99$: Share Of The Variation Of $\ln R$ Explained By The Leading Terms

Term	Fat-tree	Clos	QFly	BCube	DCell
Main effects					
Emission efficiency	64.2%	65.3%	65.7%	64.6%	64.1%
Attempt rate	10.6%	10.4%	9.2%	10.8%	11.1%
Communication-qubit cooldown time	9.5%	9.2%	8.2%	9.5%	9.9%
Switch insertion loss	4.4%	3.7%	4.1%	3.5%	3.9%
Connector loss	1.1%	0.9%	1.1%	0.9%	0.9%
Two-factor interactions					
Cooldown time × attempt rate	9.3%	9.0%	8.1%	9.5%	9.9%
Sum (of listed terms)	**99.2%**	**98.5%**	**96.4%**	**98.9%**	**99.8%**

Note: Shares of the variation of $\ln R$ *among the combination means of the* 2^{10} *combinations with* $F_0 = 0.99$*, all of which serve requests (Section VI-B); each topology is evaluated at its own* W_{sat} *(Table VII). All other terms together explain the remaining 0.2–3.6%.* F_0 *is held at 0.99 in these combinations, and* V_{HOM}*,* τ_{coh}*,* F_{swap}*,* d_{inter} *and* λ_{dark} *fall among the remaining terms. Section VI-B gives the gain of each factor. Sums are computed from unrounded values.*

undistilled pairs; longer distances or wider windows would make fiber loss and dark counts matter, and distillation would lower the fidelity a raw link needs. The OFAT curves are traced only at the favorable corner, so a factor's response when other factors are less favorable is known only through the two-level interaction terms of the DOE.

The evaluated instances hold 54–64 QPUs; Appendix E gives the rules that carry the framework to other sizes, and since we report no runs there, the paper claims a method for them, not measured results.

Q-REFINE can be extended to include circuit-derived workloads, additional detector regimes, cost- and resource-footprint normalization, and more detailed control-plane timing models. These extensions would further connect physical-device assumptions, interconnect design, and application-level execution requirements.

VIII. CONCLUSION

This paper presented Q-REFINE, a framework for design optimization of QDC interconnects. It combines a controlled inter-QPU request workload, topology instances with explicit QPUs, switches, fiber links and BSM units, a discrete-event simulator, and a three-stage study of load selection, full-factorial DOE and OFAT sweeps.

For each family, load selection sets the load, and the DOE and OFAT sweeps answer both design questions at that load. The emitted-pair fidelity, the HOM visibility and the memory coherence time decide together whether any request is served. Where requests are served, the emission efficiency changes the completion rate most, 8.4–9.5 times between its two DOE levels, followed by the attempt rate and the communication-qubit cooldown time, which act together, and then the switch insertion and connector losses. The emitted-pair fidelity and the HOM visibility pay off only up to the step the longest feasible paths need, an emitted-pair fidelity of 0.931 for one segment and 0.978 for two, and the coherence time pays off up to about 10 ms, only for paths with more than one segment. The rate was still rising at the best reported emission efficiency, 0.163, the switch insertion and connector losses pay off over their whole ranges, and a higher attempt rate pays off only until $1/f$ falls below the time needed to learn an attempt's outcome and re-initialize the qubit.

APPENDIX A VALIDATION AGAINST EVENT-LEVEL SIMULATORS

Our simulator's link and swap models cover ground that SeQUeNCe, QuISP and NetSquid also cover, so where they overlap, its results can be checked against theirs (Section II-C). Such a check needs care, because even event-level simulators differ. Chung et al. [18] find that SeQUeNCe and QuISP, given identical error parameters, agree on the fidelity of the pairs they distribute but differ in generation time by a factor of about four because of their handshakes. We therefore compare our simulator with these tools on matched cases, and we check separately the parts that no event-level simulator models.

First, the link model and its timing agree with tools that share none of our code. Our double-click acceptance probability and link fidelity equal those of the independent implementation in NetSquid's netsquid-magic package [14] to within 10^{-15} at 288 parameter settings spanning the ranges of arm transmission, dark counts, HOM visibility and emitted-pair fidelity. The acceptance probability sampled by SeQUeNCe's event-driven photon model [15] at five loss and source settings with unit HOM visibility, and by QuISP [16] at three detector efficiencies without loss or noise, contains our value in its 95% confidence interval. Reduced to a single link, our simulator reproduces the exact distribution of the time to the first heralded pair at four parameter settings, as netsquid-magic's sampler of the number of attempts to success does with the same attempt interval; this checks the per-attempt probability and the geometric distribution of attempts rather than the attempt interval itself. SeQUeNCe does not model a herald from one photon and a dark count, so at a dark-count rate of 1 MHz its acceptance probability is about half of ours; at the rates of our range, 1 kHz or less, this herald adds 0.07–0.10% to the acceptance probability.

Second, the best-case fidelity $\hat{F}$ of Eq. (3), which decides

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

whether a request is infeasible, errs by less than 0.01 near $F_{\min}$. It composes a path's fidelity from scalar link fidelities, which for the Bell-diagonal states of our links never exceeds the fidelity of exact composition. Against NetSquid runs that swap full two-qubit states, it underestimates the fidelity by at most 0.005 near $F_{\min}$, and the scalar decay law for stored pairs is the only term that can err the other way, by at most 0.006 near $F_{\min}$. At the favorable corner, exact composition moves no multi-segment request of BCube or DCell between feasible and infeasible, but next to the V_{HOM} step it does. At $V_{HOM} = 0.909$ a two-segment path reaches 0.852 under exact composition against 0.849 under the product rule, so under exact composition the step of Fig. 5 would lie one grid point lower, at 0.909 instead of 0.918; the F_0 step lies at 0.98 under both rules.

Third, the model of shared communication qubits, BSM units and switch ports (Section III-E), which event-level simulators do not include, was checked in small test cases whose timing can be computed by hand and under load. In the test cases, requests sharing one communication qubit finished at the computed times to the picosecond, a second request waited for a BSM unit or a switch port until the first released it, and a stored link's resources were released at its herald and reserved again after the link expired. Under load, in all five topologies, every allocated communication qubit, switch port and BSM unit was accounted for at each of 2,634 events.

APPENDIX B LINK MODEL

This appendix gives the parts of the link model of Section III-F that the main text states without derivation, in the order of that section.

A. WHY DOUBLE-CLICK HERALDING

Double-click heralding needs only that the qubits emit photons [30], and it has been demonstrated between remote NV centers [31], neutral atoms [32] and trapped ions [33]. We use it rather than single-click heralding, which is also emission-based but heralds a pair on one detected photon. In single-click, photon loss turns one qubit state into the other, so loss lowers fidelity directly, and the optical path length must be stabilized [30]. In time-bin double-click, where each photon carries its qubit in an early or a late time bin, loss lowers only the rate, and the optical phase needs to be stable only over the spacing of the time bins. A switched fabric gives each request a different optical path, so double-click avoids stabilizing every path.

B. FREQUENCY CONVERSION

Conversion enters only through η_{emit}, Eq. (5), since we neglect its noise (Section III-B); the simulator's link model has a term for converter noise, which is zero in every run. The resulting upper bound on F_{link} is tight when $\eta_A\eta_B$ is large and loose when it is small, because converter noise, like dark counts, matters relative to the signal $\eta_A\eta_B$ it competes with, and so matters more when that signal is weak.

C. CLICK RECORDS, ACCEPTANCE PROBABILITY AND LINK FIDELITY

Published sources report the fidelity F_0 of the emitted qubit–photon pair, but not its error structure, so we model the source as a Werner channel. With probability F_0 this channel leaves the emitted pair intact; otherwise it turns the pair into each of the other three Bell states with equal probability. To check this choice, we repeated the derivation below for a general Pauli source, which turns the pair into the other three Bell states with any probabilities that sum to $1 - F_0$. With $V_{HOM} = 1$ and no dark counts, the Werner channel gives the lowest F_{link} of all such sources. For $V_{HOM} \geq 0.8$ and $P_{dark} \leq 10^{-3}$, it gives a lower F_{link} than pure dephasing, which applies only phase flips, and lies within 0.004 of the lowest, which errors that shift the photon's time bin reach.

Each photon reaches a detector with probability $x = \eta_A\eta_{det}$ or $y = \eta_B\eta_{det}$. The two photons meet on a balanced beam splitter, where the fraction V_{HOM} of their overlap interferes and the rest behaves as distinguishable light, and each of the four detector–bin pairs registers a dark click with probability $p_d = P_{dark}$ (Section III-F). A click record c is the set of detector–bin pairs that click. If S is the set that the photons alone would fire, the record occurs with probability $\mathbf{1}[S \subseteq c]\, p_d^{|c\setminus S|}(1-p_d)^{4-|c|}$, and ρ_c is the mixture of the photon-only states weighted by these probabilities. The per-attempt acceptance probability and the link fidelity of segment σ are

$$P_{acc}(\sigma) = \sum_{c\in\mathcal{A}} \Pr(c), \tag{B.1}$$

$$F_{link}(\sigma) = \frac{1}{P_{acc}(\sigma)} \sum_{c\in\mathcal{A}} \Pr(c)\, \langle\Psi_c|\rho_c|\Psi_c\rangle, \tag{B.2}$$

where $\Pr(c)$ is the probability above averaged over the photon outcomes S, $\mathcal{A}$ is the set of the four accepted records, and Ψ_c is the Bell state that record c heralds (Section III-F). Evaluating these sums gives, with $V = V_{HOM}$,

$$P_{acc} = \tfrac{1}{2}(1-p_d)^2 D, \qquad F_{link} = \frac{N}{18\,D}, \tag{B.3}$$

where

$$\begin{aligned} D &= xy\,[1 - p_d(7-V)] + 4p_d(x+y) \\ &\quad + 8p_d^2(1-x)(1-y), \\ N &= xy\,(k_1 - p_d\,k_2) + 18p_d(x+y) \\ &\quad + 36p_d^2(1-x)(1-y), \\ k_1 &= G(2V+1) + V + 5, \quad k_2 = G(V+1) + 32 - 4V, \\ G &= 4F_0(2F_0 - 1). \end{aligned}$$

In D, the first term is a coincidence of the two photons, the second one photon and one dark click, and the third two dark clicks. At $p_d = 0$, $P_{acc} = xy/2$, and with $F_0 = V_{HOM} = 1$ also $F_{link} = 1$; at $x = y = 0$, $F_{link} = 1/4$. Between the

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

two, the fidelity is set by the share of false heralds, which grows with P_{dark} relative to $\eta_A\eta_B$. We derived Eq. (B.3) symbolically from the Werner source, arm loss, the beam splitter and dark counts, and the simulator's link model agrees with it to within 10^{-14} at 400 random parameter points.

D. DETECTOR RECOVERY

After a click, a detector needs a dead time $T_{dead} = 58$ ns to recover, and the station accepts no new attempt until both of its detectors are ready. An attempt with a click therefore lasts at least until the dead time after its last click has passed; this lengthens the attempt interval of Eq. (7) only when that interval is shorter than the time from the attempt's start to the detectors' recovery. A generation round, Eq. (10), ends at T_{gen} after its start, so an attempt that would not finish within it is left to the next round.

E. STORAGE

We omit amplitude damping from the memory model, because it is much slower than dephasing in the memories we consider. In Eq. (8), t_e is the emission time of the successful attempt, so the decay includes the wait for the herald. Eq. (8) is exact for pairs whose errors are phase flips and slightly optimistic for Werner pairs; at $F_{link} = 0.9$ and $t - t_e = 0.1\,\tau_{coh}$, for example, it gives 0.862 against 0.859.

F. ENTANGLEMENT SWAPPING

The swap fidelity F_{swap} of Eq. (9) includes the swap's gate, readout and feed-forward errors. For Werner inputs the exact composition is $F_L F_R + (1 - F_L)(1 - F_R)/3$ before swap noise, so the product rule slightly understates it. For two links at 0.8, it gives 0.640 instead of 0.653. The swap rule alone therefore understates multi-segment fidelities; with the slightly optimistic storage model of Eq. (8), the net error is small and can go either way.

APPENDIX C EVALUATION SETTINGS

This appendix gives the argument for the load at which each topology is run (Section III-B), the constraints that define a request batch (Section III-D), the robustness of load selection to the deadline (Section V-A) and the basis of the fidelity threshold (Section V-B).

A. THE TWO PARTS OF A MEASURED EFFECT

At a fixed W, the change ΔR in completion rate when a factor is improved then has two parts,

$$\Delta R = \Delta R_{comp} + \Delta R_{cong}, \tag{C.1}$$

where ΔR_{comp} is the gain from the better component itself and ΔR_{cong} is the rate won back from congestion, because a better component also lets each request release its resources sooner. A run measures only ΔR, and ΔR_{cong} grows with W. Running at W_{sat} gives up at most 5% of that rate and in return keeps ΔR_{cong} as small as the saturation criterion allows. The 5% is our choice rather than a sourced value.

B. WORKLOAD CONSTRAINTS

The workload generator of Section III-D builds a batch of M requests from the endpoint pool Q_{eff}. For request i, let s_i and d_i denote its source and destination QPUs. The sampled request batch satisfies

$$\begin{aligned}(s_i,\ d_i), \quad & i = 1, \ldots, M, \\ & s_i, d_i \in Q_{eff}, \quad s_i \neq d_i.\end{aligned} \tag{C.2}$$

Let C_{ep} denote the endpoint-use limit. For the balanced batches used in this study, each selected endpoint appears C_{ep} times, so that

$$|\{\, i : s_i = q \,\}| + |\{\, i : d_i = q \,\}| = C_{ep}, \quad q \in Q_{eff}. \tag{C.3}$$

This balance implies

$$2M = |Q_{eff}|\ C_{ep}. \tag{C.4}$$

We set $C_{ep} = n_{comm}$, which gives Eq. (2). When the product $|Q_{eff}|\, n_{comm}$ is odd, one endpoint appears $n_{comm} - 1$ times instead of C_{ep}. During execution, the simulator still enforces communication-qubit availability through resource reservation.

Let g(q) denote the rack containing QPU q. The parameter p_{inter} sets the target fraction of requests whose two endpoints are in different racks, so that

$$|\{\, i : g(s_i) \neq g(d_i) \,\}| = \left\lfloor M p_{inter} + \tfrac{1}{2} \right\rfloor. \tag{C.5}$$

The remaining request pairs are drawn from endpoints in the same rack while maintaining the endpoint-use limit.

C. ROBUSTNESS TO THE SERVICE DEADLINE

Load selection was repeated at $T_D = 259$ μs and 2 ms (Section V-A). At 259 μs the 5% limit on unserved requests admits no W for Fat-tree, Clos, QFly and BCube, whose smallest unserved shares over the W grid are 8.1%, 6.8%, 6.9% and 6.3%; the unserved requests are those whose generation alone takes longer than the deadline. At 259 μs a 10% limit on unserved requests admits $W_{sat} = 41$, 81, 108 and 68 for these four topologies, and DCell has $W_{sat} = 20$ under the 5% limit and 27 under the 10% limit, with no W under the 2% limit. At 2 ms no request goes unserved at any W, so the 2%, 5% and 10% limits select the same W_{sat}, 41, 54, 54, 54 and 27 for Fat-tree, Clos, QFly, BCube and DCell. R peaks at $W = 54$, 54, 108, 68 and 27 and then falls, because batches last longer as more requests wait for resources (Fat-tree, from $W = 54$ to 135: batch duration 756 to 896 μs, share of requests that waited 0.25 to 0.52). The selected load therefore depends on the deadline, which a study must state before it evaluates designs.

D. FIDELITY THRESHOLD

No single fidelity makes a pair usable. The requirement depends on the application, on whether error correction or entanglement distillation (turning several noisy pairs into fewer better ones) is used, and on how many non-local operations a circuit uses (Table IX). Section V-B takes the

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

TABLE IX. Minimum Bell-pair fidelity by use of the delivered pair, from published requirements

Use of the pair	Minimum F	Basis
Teleportation better than any classical channel	> 0.5	average teleportation fidelity $(2F+1)/3$ above the classical $2/3$ [35]
Verifiable blind quantum computation	0.786–0.808	teleportation-fidelity targets 0.857–0.872 [22], converted by the relation above (ours)
Entanglement-based key distribution	> 0.835	bit and phase error rates below 11% [36]; Werner error rate $2(1-F)/3$ (ours)
Modular fault-tolerant computation, no distillation	**about 0.85**	**about 10% interface flip probability [27]; Werner conversion (ours); used here**
Surface-code repeater chain, no purification	> 0.96 (≥ 0.92 at low herald success)	link fidelity that surface-code communication tolerates [37]
Distributed algorithm without error correction	0.97 (measured, not a minimum)	pairs at 0.9689 gave a teleported CZ gate of 86% fidelity [33]

requirement of modular fault-tolerant computation, about 1% error inside each module and about 10% bit- and phase-flip probability at the interface [27], which is $F = 0.85$ for a Werner pair.

Three qualifications apply. First, at threshold a larger code no longer suppresses logical errors, so practical operation needs links well below 10% error. Second, the 10% maps to a fidelity only through a noise model. For other Bell-diagonal pairs, i.e., other mixtures of the four Bell states, the same 10% needs F between 0.80 and 0.90, and local gate noise at the interface uses part of the budget. The heralded pair of Appendix B-C is not exactly a Werner pair, because imperfect HOM visibility adds phase flips only, so at $F = 0.85$ its phase-flip probability is somewhat above 10%, and $F_{\min} = 0.85$ is the threshold of a Werner pair of equal fidelity. Storage adds only phase flips; at the favorable corner, under the reservation rules of Section III-E, it raises the phase-flip probability by at most 3×10^{-6} above that of a Werner pair, because no pair is stored longer than T_D against a coherence time of 75 s; at short coherence times stored pairs carry more phase flips, so for those designs 0.85 is slightly optimistic. Third, where distillation is available, a lower raw-link threshold applies [34].

APPENDIX D SIMULATOR MODULES

Table X lists the modules of the simulator of Section IV in two groups. A design enters only through the first; the second, including the request controller, is the same for every design and topology, which lets one controller serve all five topologies. Each module has a defined input and output, so its implementation can change without changing its callers. The request controller plays the role of the link-layer protocol of Avis et al. [22], which calls the link-layer service of [38].

TABLE X. Simulator modules, the part of Section III each implements, and their inputs and outputs

Module (Sec.)	Inputs, outputs and role
Depend on the design	
Topology and resources (III-C)	G, reserve(π) $\rightarrow$ granted or refused. Builds the topology instance G and holds $\mathcal{R}$, its communication qubits (n_{comm} per QPU), BSM units and ports; reserves and releases them.
Routing (III-E)	$(r, G, \theta) \rightarrow \Pi_r$ with $\hat{F}(\pi)$. Gives each request its candidate paths Π_r, pruned by Eq. (3); a path π is a chain of segments σ.
Link generation (III-F)	$(\sigma, \theta) \rightarrow$ herald time, $F_{link}(\sigma)$, Eq. (5), Eqs. (B.1)–(B.2).
Storage and swapping (III-F)	(two stored links, t) $\rightarrow$ joined link and its fidelity, Eqs. (8), (9).
Independent of the design	
Workload builder (III-D)	Seed $\rightarrow \mathcal{B}$. Draws the batch $\mathcal{B}$ of M requests $r = (s_r, d_r)$, Eq. (2) and Appendix C-B.
Event core	Handles events in time order, simultaneous events in the order they were scheduled.
Request controller (III-E)	Keeps at most W requests in $A(t)$, Eq. (4); ends each request once, as served, unserved or infeasible.
Metrics recorder (III-B)	$F_{end}(r)$ and $\mathrm{Accept}(r)$ against $F_{\min}$ (Section III-F); the completion rate $R_{F_{\min}}$, Eq. (1); the unserved and infeasible shares.
Experiment driver (V)	Runs the stages of Fig. 1; one run per design, concurrency limit and seed; one record per run.

APPENDIX E OTHER SIZES

At another size the construction parameters change, and four settings follow from them by stated rules. Our code builds the five families at nominal scales 16, 64 and 128, a nominal scale being the QPU count each construction approximates; the instances of Table III, with 54–64 QPUs, are those of scale 64. The physical models do not change, because they are defined per link and per node. First, the workload keeps the formulation of Section III-D with a new pool, whose size $|Q_{eff}|$ is the QPU count of the smallest construction at that size (16 and 125), so Eq. (2) gives $M = 40$ and 312. Second, load selection sets the concurrency limit W again for each topology and size, because the load at which a network saturates depends on its size. Third, the total B of BSM units keeps its rule of one unit per two endpoint communication qubits, so $B = 40$ and 312. Fourth, the construction sets path length in the server-centric topologies. A one-level BCube (scale 64) has paths of at most two segments and a two-level BCube (scale 128) up to three; a one-level DCell has up to three at every size used here. At scale 128, 23 of the 312 BCube requests and 80 of the 312 DCell requests need three segments. Because every segment and swap lowers the best-case fidelity of Eq. (3), these shares bound, before any run and at given factor levels, the share of requests that can meet $F_{\min}$. With one fiber per communication qubit, static port use stays within the 128-port switches at all three scales, at most 76 of 128.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

APPENDIX F FACTOR RANGES, SOURCES AND DOE LEVELS

Table XI gives the source of each end of each factor's reported range. A range runs from the least favorable demonstrated value to the most favorable value reported, demonstrated or projected; an end is marked projected when its source measured it under conditions that a network link lacks or states it as a design value.

The DOE runs each factor at the favorable end of its range and at a less favorable level (Table VI). For η_{emit}, f and F_0 the level follows the rule of Section V-D, giving 0.0505, 0.161 MHz and 0.94; for the other eight it is the less favorable end of the range. Each OFAT sweep covers the whole range, except for these three factors, whose sweeps start one grid step below the value at which the rate falls under 1% of the favorable-corner rate, at 0.00485, 0.00237 MHz and 0.92.

Two ends differ from their sources. The runs used 75 s as the favorable coherence time, whereas the source reports at most 63 s; above about 10 ms the rate no longer changes (Section VI-C), so the difference changes no result. The less favorable cooldown time, 5.8 μs, adds a 2 μs initialization [39] to a 3.8 μs attempt [40] that may already include part of the initialization, so the demonstrated value may be shorter and the gain of Δt_{cool} is an upper estimate.

APPENDIX G ADDITIONAL RESULTS

Full design. Over all 2048 combinations of the 2^{11} design, main effects explain 42–51% of the variation of R, two-factor interactions 31–34% and higher orders 18–24%, and F_0 has the second-largest effect after η_{emit} in every topology. These large interaction shares come from the 768 combinations that serve no request (Section VI-B), which is why Section V-D analyzes the design in two steps.

Resources at W_{sat}. In Fat-tree, Clos and QFly all 135 requests of a batch use one-segment paths; BCube has 102 one-segment and 33 two-segment requests, and DCell 95, 3 and 37 with one, two and three segments, in one batch; the counts vary slightly between seeds. Over 30 batches the share of requests that waited for a resource is 25%, 14%, 9%, 37% and 46% for Fat-tree, Clos, QFly, BCube and DCell. Fat-tree, QFly and DCell waited mostly for BSM units (752 of 1005, 372 of 375 and 1871 of 1912 waits), Clos mostly for switch ports (407 of 581) and BCube mostly for communication qubits (7635 of 8460). The mean batch lasts 1.01–1.36 times T_D.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

TABLE XI. Sources Of The Factor Ranges And Of The Fixed Settings

Factor	End	Status	Source	Note
Swept factors (Table VI), less favorable end first				
F_0	0.65	demonstrated	[41]	measured lower bound
	0.99	projected	[22], [42]	measured 0.974 corrected for detector dark counts in [22], with conversion taken as ideal
V_{HOM}	0.89	demonstrated	[22], [43]	model value at a 0.5 μs coincidence window, fitted to the data of [43]
	0.955	projected	[32]	contrast at zero delay within a 70 ns acceptance window
τ_{coh}	0.30 μs	demonstrated	[44]	under dynamical decoupling in an emitting node (preprint)
	63 s	projected	[45]	isolated register qubit under dynamical decoupling; the runs used 75 s (see text)
F_{swap}	0.95	demonstrated	[22], [46]	two-qubit gate fidelity, taken as the swap fidelity
	0.97	demonstrated	[22], [47]	two-qubit gate fidelity, taken as the swap fidelity
η_{emit}	2.59×10^{-4}	demonstrated	[40], [48], [49]	composed as 5.1×10^{-4} detection probability / 0.95 detector efficiency $\times$ 0.48 conversion efficiency, which gives 2.58×10^{-4} from these rounded values
	0.163	demonstrated	[50], [51]	composed as $0.69 \times 0.97 \times 0.81$ to the fiber $\times$ 0.30 conversion, from two runs of one apparatus
f	10^{-4} MHz	demonstrated	[52]	repetition rate of a qubit-to-optical transduction experiment
	10.9 MHz	demonstrated	[53]	quoted attempt rate
Δt_{cool}	5.8 μs	demonstrated	[22], [39], [40]	sum of a 3.8 μs attempt and a 2 μs initialization, compiled in [22]; may be longer than demonstrated (see text)
	0.83 μs	demonstrated	[33]	sum of 7 ns emission and 820 ns state preparation, from Extended Data Fig. 3
L_{sw}	1.5 dB	demonstrated	[54]	typical value of a 384-port switch
	0.35 dB	projected	[55]	typical value of a 16-port switch; our switches have 128 ports
L_{conn}	0.75 dB	demonstrated	[56], [57]	standard-grade allowance for a mated pair
	0.2 dB	projected	[56], [57]	reference-grade mated pair
d_{inter}	100 m	demonstrated	[1], [58]	fixed distance of a benchmark study, equal to the reach of short-reach optical Ethernet
	3 m	projected	[59], [60]	1 m rack width plus 2 m cable overhead, from a data-center cost model
λ_{dark}	10^3 Hz	demonstrated	[61]	at 93% system detection efficiency
	10^{-4} Hz	projected	[62]	at 1.5% system detection efficiency, because of cold spectral filtering
Fixed settings (Table V)				
η_{det}	0.98	demonstrated	[63]	fiber-coupled, at 1550 nm
T_{dead}	58 ns	demonstrated	[64]	fitted $1/e$ time constant of the bias-current recovery
α_f	0.18 dB/km	demonstrated	[65]	specified maximum at 1550 nm
d_{intra}	2 m	demonstrated	[33]	separation of two linked modules
Δt_{cw}	15 ns	demonstrated	[48]	default detection window of a heralded link
Derived value (Section II-C)				
P_{acc}	3.2×10^{-8}	derived	[40], [48], [49], [63]	$\frac{1}{2}\eta_{emit}^2\eta_{det}^2$ at the less favorable η_{emit} and $\eta_{det} = 0.98$

Note: An end is marked projected when its source measured it under conditions that a network link lacks, such as an isolated qubit, a post-selected window or a narrow filter, or states it as a design value. A composed end multiplies or adds quantities reported in the cited sources.

REFERENCES

[1] S. Pouryousef, E. Kaur, H. Shapourian, D. Towsley, R. Kompella, and R. Nejabati, "Benchmarking quantum data center architectures: A performance and scalability perspective," arXiv preprint arXiv:2601.01353, 2026. [Online]. Available: https://arxiv.org/abs/2601.01353

[2] H. Shapourian, E. Kaur, T. Sewell, J. Zhao, M. Kilzer, R. Kompella, and R. Nejabati, "Quantum data center infrastructures: A scalable architectural design perspective," arXiv preprint arXiv:2501.05598, 2025. [Online]. Available: https://arxiv.org/abs/2501.05598

[3] A. S. Cacciapuoti, C. Pellitteri, J. Illiano, L. d'Avossa, F. Mazza, S. Chen, and M. Caleffi, "Quantum data centers: Why entanglement changes everything," arXiv preprint arXiv:2506.02920, 2025. [Online]. Available: https://arxiv.org/abs/2506.02920

[4] D. Sakuma, T. Tsuno, H. Shimizu, Y. Kurosawa, M. T. Friedrich, K. Teramoto, A. Taherkhani, A. Todd, Y. Ueno, M. Hajdušek, R. Ikuta, R. V. Meter, T. Sasaki, and S. Nagayama, "Q-Fly: An optical interconnect for modular quantum computers," arXiv preprint arXiv:2412.09299, 2024. [Online]. Available: https://arxiv.org/abs/2412.09299

[5] Y. Xin and L. Zhang, "Towards a high-performance quantum data center network architecture," in 2025 IEEE International Conference on Communications (ICC), 2025, pp. 2400–2405.

[6] Z. Yang, C. Tian, R. Jain, R. Kompella, R. Nejabati, M. Hamdi, A. Erbad, and H. Shapourian, "Effective scheduling for quantum data centers," in Proceedings of IEEE Quantum Week 2025 (QCE 2025), vol. 2, 2025, pp. 454–455.

[7] H. Zhang, Y. Xu, H. Hu, K. Yin, H. Shapourian, J. Zhao, R. R. Kompella, R. Nejabati, and Y. Ding, "SwitchQNet: Optimizing distributed quantum computing for quantum data centers with switch networks," in Proceedings of the 52nd Annual International Symposium on Computer Architecture, ser. ISCA '25, 2025, pp. 1449–1463.

[8] H. Zhang, Y. Xu, H. Hu, K. Yin, H. Shapourian, J. Zhao, R. R. Kompella, R. Nejabati, and Y. Ding, "Optimizing quantum communication for quantum data centers with reconfigurable networks," arXiv preprint arXiv:2412.04837, 2024. [Online]. Available: https://arxiv.org/abs/2412.04837

[9] S. Pouryousef, R. Nejabati, D. Towsley, R. Kompella, and E. Kaur, "Network-aware scheduling for remote gate execution in quantum data centers," arXiv preprint arXiv:2504.20176, 2025. [Online]. Available:

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

https://arxiv.org/abs/2504.20176

[10] E. Kaur, H. Shapourian, J. Zhao, M. Kilzer, R. Kompella, and R. Nejabati, "Optimized quantum circuit partitioning across multiple quantum processors," arXiv preprint arXiv:2501.14947, 2025. [Online]. Available: https://arxiv.org/abs/2501.14947

[11] F. Burt, K.-C. Chen, and K. K. Leung, "Generalized circuit partitioning for distributed quantum computing," in Proceedings of the 2024 IEEE International Conference on Quantum Computing and Engineering (QCE), 2024.

[12] A. Wu, H. Zhang, G. Li, A. Shabani, Y. Xie, and Y. Ding, "AutoComm: A framework for enabling efficient communication in distributed quantum programs," in Proceedings of the 55th Annual IEEE/ACM International Symposium on Microarchitecture, ser. MICRO '22, 2022, pp. 1027–1041.

[13] K.-C. Chen, F. Burt, N. K. Panigrahy, and K. K. Leung, "Adaptive resource orchestration for distributed quantum computing systems," IEEE Internet Computing, vol. 30, no. 1, pp. 16–25, 2026.

[14] T. Coopmans, R. Knegjens, A. Dahlberg, D. Maier, L. Nijsten, J. de Oliveira Filho, M. Papendrecht, J. Rabbie, F. Rozpędek, M. Skrzypczyk, L. Wubben, W. de Jong, D. Podareanu, A. Torres-Knoop, D. Elkouss, and S. Wehner, "NetSquid, a NETwork simulator for Quantum information using discrete events," Communications Physics, vol. 4, no. 1, p. 164, 2021.

[15] 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 Science and Technology, vol. 6, no. 4, p. 045027, 2021.

[16] R. Satoh, M. Hajdušek, N. Benchasattabuse, S. Nagayama, K. Teramoto, T. Matsuo, S. A. Metwalli, T. Satoh, S. Suzuki, and R. V. Meter, "QuISP: a quantum internet simulation package," in Proceedings of the 2022 IEEE International Conference on Quantum Computing and Engineering (QCE), 2022, pp. 353–364.

[17] L. Chen, K. Xue, J. Li, N. Yu, R. Li, Q. Sun, and J. Lu, "SimQN: A network-layer simulator for the quantum network investigation," IEEE Network, vol. 37, no. 5, pp. 182–189, 2023.

[18] J. Chung, M. Hajdušek, N. Benchasattabuse, A. Kolar, A. Singal, K. S. Soon, K. Teramoto, A. Zang, R. Kettimuthu, and R. V. Meter, "Cross-validating quantum network simulators," in IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2025.

[19] S. Zhang, L. Xiong, Y. Liu, B. L. Mark, L. Yang, Z. Yang, and W. Jiang, "An end-to-end distributed quantum circuit simulator," arXiv preprint arXiv:2511.19791, 2025. [Online]. Available: https://arxiv.org/abs/2511.19791

[20] R. Jain, The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling. John Wiley & Sons, 1991.

[21] D. Ferrari and M. Amoretti, "A design framework for the simulation of distributed quantum computing," arXiv preprint arXiv:2306.11539, 2023. [Online]. Available: https://arxiv.org/abs/2306.11539

[22] G. Avis, F. Ferreira da Silva, T. Coopmans, A. Dahlberg, H. Jirovská, D. Maier, J. Rabbie, A. Torres-Knoop, and S. Wehner, "Requirements for a processing-node quantum repeater on a real-world fiber grid," npj Quantum Information, vol. 9, p. 100, 2023.

[23] D. C. Montgomery, Design and Analysis of Experiments, 10th ed. John Wiley & Sons, 2019.

[24] M. Iuliano, N. Demetriou, H. B. van Ommen, C. Karels, T. H. Taminiau, and R. Hanson, "Unconditionally teleported quantum gates between remote solid-state qubit registers," Nature Communications, vol. 17, p. 4694, 2026.

[25] S. D. Barrett and P. Kok, "Efficient high-fidelity quantum computation using matter qubits and linear optics," Physical Review A, vol. 71, no. 6, p. 060310, 2005.

[26] C. Jones, D. Kim, M. T. Rakher, P. G. Kwiat, and T. D. Ladd, "Design and analysis of communication protocols for quantum repeater networks," New J. Phys., vol. 18, p. 083015, 2016.

[27] J. Ramette, J. Sinclair, N. P. Breuckmann, and V. Vuletić, "Fault-tolerant connection of error-corrected qubits with noisy links," npj Quantum Information, vol. 10, p. 58, 2024.

[28] W. J. Yun, D. Nag, S. Ballabh, J. Zhao, E. Kaur, and P. Das, "ATHENA: A compiler for optimized scheduling in distributed quantum computers," arXiv:2605.21795, 2026.

[29] J. Sinclair, J. Ramette, B. Grinkemeyer, D. Bluvstein, M. D. Lukin, and V. Vuletić, "Fault-tolerant optical interconnects for neutral-atom arrays," Physical Review Research, vol. 7, p. 013313, 2025.

[30] H. K. C. Beukers, M. Pasini, H. Choi, D. Englund, R. Hanson, and J. Borregaard, "Remote-entanglement protocols for stationary qubits with photonic interfaces," PRX Quantum, vol. 5, p. 010202, 2024.

[31] B. Hensen et al., "Loophole-free Bell inequality violation using electron spins separated by 1.3 kilometres," Nature, vol. 526, pp. 682–686, 2015.

[32] T. van Leent, M. Bock, F. Fertig, R. Garthoff, S. Eppelt, Y. Zhou, P. Malik, M. Seubert, T. Bauer, W. Rosenfeld, W. Zhang, C. Becher, and H. Weinfurter, "Entangling single atoms over 33 km telecom fibre," Nature, vol. 607, pp. 69–73, 2022.

[33] D. Main, P. Drmota, D. P. Nadlinger, E. M. Ainley, A. Agrawal, B. C. Nichol, R. Srinivas, G. Araneda, and D. M. Lucas, "Distributed quantum computing across an optical network link," Nature, vol. 638, pp. 383–388, 2025.

[34] N. H. Nickerson, Y. Li, and S. C. Benjamin, "Topological quantum computing with a very noisy network and local error rates approaching one percent," Nature Communications, vol. 4, p. 1756, 2013.

[35] M. Horodecki, P. Horodecki, and R. Horodecki, "General teleportation channel, singlet fraction, and quasidistillation," Physical Review A, vol. 60, pp. 1888–1898, 1999.

[36] P. W. Shor and J. Preskill, "Simple proof of security of the BB84 quantum key distribution protocol," Physical Review Letters, vol. 85, pp. 441–444, 2000.

[37] A. G. Fowler, D. S. Wang, C. D. Hill, T. D. Ladd, R. Van Meter, and L. C. L. Hollenberg, "Surface code quantum communication," Physical Review Letters, vol. 104, p. 180503, 2010.

[38] A. Dahlberg, M. Skrzypczyk, T. Coopmans, L. Wubben, F. Rozpędek, M. Pompili, A. Stolk, P. Pawełczak, R. Knegjens, J. de Oliveira Filho, R. Hanson, and S. Wehner, "A link layer protocol for quantum networks," in Proceedings of the ACM Special Interest Group on Data Communication (SIGCOMM '19), 2019, pp. 159–173.

[39] A. Reiserer, N. Kalb, M. S. Blok, K. J. M. van Bemmelen, T. H. Taminiau, R. Hanson, D. J. Twitchen, and M. Markham, "Robust quantum-network memory using decoherence-protected subspaces of nuclear spins," Physical Review X, vol. 6, p. 021040, 2016.

[40] M. Pompili, S. L. N. Hermans, S. Baier, H. K. C. Beukers, P. C. Humphreys, R. N. Schouten, R. F. L. Vermeulen, M. J. Tiggelman, L. dos Santos Martins, B. Dirkse, S. Wehner, and R. Hanson, "Realization of a multinode quantum network of remote solid-state qubits," Science, vol. 372, pp. 259–264, 2021.

[41] N. Coste et al., "High-rate entanglement between a semiconductor spin and indistinguishable photons," Nature Photonics, vol. 17, pp. 582–587, 2023.

[42] A. Stute, B. Casabone, P. Schindler, T. Monz, P. O. Schmidt, B. Brandstätter, T. E. Northup, and R. Blatt, "Tunable ion–photon entanglement in an optical cavity," Nature, vol. 485, pp. 482–485, 2012.

[43] V. Krutyanskiy et al., "Entanglement of trapped-ion qubits separated by 230 meters," Physical Review Letters, vol. 130, p. 050803, 2023.

[44] H. Huet et al., "Dynamical decoupling of a quantum dot spin in a micropillar cavity for spin-multiphoton entanglement," arXiv:2606.28535, 2026.

[45] C. E. Bradley, J. Randall, M. H. Abobeih, R. C. Berrevoets, M. J. Degen, M. A. Bakker, M. Markham, D. J. Twitchen, and T. H. Taminiau, "A ten-qubit solid-state spin register with quantum memory up to one minute," Physical Review X, vol. 9, p. 031045, 2019.

[46] V. Krutyanskiy, M. Canteri, M. Meraner, J. Bate, V. Krcmarsky, J. Schupp, N. Sangouard, and B. P. Lanyon, "Telecom-wavelength quantum repeater node based on a trapped-ion processor," Physical Review Letters, vol. 130, p. 213601, 2023.

[47] N. Kalb, A. A. Reiserer, P. C. Humphreys, J. J. W. Bakermans, S. J. Kamerling, N. H. Nickerson, S. C. Benjamin, D. J. Twitchen, M. Markham, and R. Hanson, "Entanglement distillation between solid-state quantum network nodes," Science, vol. 356, pp. 928–932, 2017.

[48] S. L. N. Hermans, M. Pompili, H. K. C. Beukers, S. Baier, J. Borregaard, and R. Hanson, "Qubit teleportation between non-neighbouring nodes in a quantum network," Nature, vol. 605, pp. 663–668, 2022.

[49] A. J. Stolk et al., "Metropolitan-scale heralded entanglement of solid-state qubits," Science Advances, vol. 10, p. eadp6442, 2024.

[50] J. Schupp, V. Krcmarsky, V. Krutyanskiy, M. Meraner, T. E. Northup, and B. P. Lanyon, "Interface between trapped-ion qubits and traveling photons with close-to-optimal efficiency," PRX Quantum, vol. 2, p. 020331, 2021.

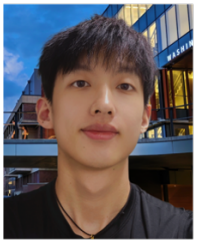
[51] V. Krutyanskiy, M. Canteri, M. Meraner, V. Krcmarsky, and B. P. Lanyon, "Multimode ion-photon entanglement over 101 kilometers," PRX Quantum, vol. 5, p. 020308, 2024.

[52] M. Mirhosseini, A. Sipahigil, M. Kalaee, and O. Painter, "Superconducting qubit to optical photon transduction," Nature, vol. 588, pp. 599–603, 2020.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686

[53] R. Stockill, M. J. Stanley, L. Huthmacher, E. Clarke, M. Hugues, A. J. Miller, C. Matthiesen, C. Le Gall, and M. Atatüre, "Phase-tuned entangled state generation between distant spin qubits," Physical Review Letters, vol. 119, p. 010503, 2017.
[54] HUBER+SUHNER Polatis, "POLATIS Series 7000n network optical switch up to 384×384 ports," data sheet, Rev. 7000n.122018.001, 2018.
[55] HUBER+SUHNER Polatis, "POLATIS Series 6000 Ultra Q quantum grade optical switch," preliminary data sheet, Rev. 6000 Ultra Q 112022.001, 2022.
[56] Optical Fiber Cabling and Components Standard, ANSI/TIA-568.3-D, Telecommunications Industry Association, 2016.
[57] Fluke Networks, "New loss budget values for reference grade connectors in ANSI/TIA-568.3-D." [Online]. Available: https://www.flukenetworks.com/content/new-loss-budget-values-reference-grade-connectors-ansitia-5683-d (accessed Sep. 25, 2026).
[58] IEEE Standard for Ethernet, Amendment 3: Physical Layer Specifications and Management Parameters for 40 Gb/s and 100 Gb/s Operation over Fiber Optic Cables, IEEE Std 802.3bm-2015, 2015.
[59] M. Besta and T. Hoefler, "Slim Fly: A cost effective low-diameter network topology," in Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC14), 2014, pp. 348–359.
[60] J. Kim, W. J. Dally, and D. Abts, "Flattened butterfly: A cost-efficient topology for high-radix networks," in Proceedings of the 34th Annual International Symposium on Computer Architecture (ISCA '07), 2007, pp. 126–137.
[61] F. Marsili, V. B. Verma, J. A. Stern, S. Harrington, A. E. Lita, T. Gerrits, I. Vayshenker, B. Baek, M. D. Shaw, R. P. Mirin, and S. W. Nam, "Detecting single infrared photons with 93% system efficiency," Nature Photonics, vol. 7, pp. 210–214, 2013.
[62] H. Shibata, K. Shimizu, H. Takesue, and Y. Tokura, "Ultimate low system dark-count rate for superconducting nanowire single-photon detector," Optics Letters, vol. 40, pp. 3428–3431, 2015.
[63] D. V. Reddy, R. R. Nerem, S. W. Nam, R. P. Mirin, and V. B. Verma, "Superconducting nanowire single-photon detectors with 98% system detection efficiency at 1550 nm," Optica, vol. 7, pp. 1649–1653, 2020.
[64] C. Autebert, G. Gras, E. Amri, M. Perrenoud, M. Caloz, H. Zbinden, and F. Bussières, "Direct measurement of the recovery time of superconducting nanowire single-photon detectors," Journal of Applied Physics, vol. 128, p. 074504, 2020.
[65] Corning Incorporated, "Corning SMF-28 Ultra optical fiber," product information sheet PI1424, Sep. 2019.



CHENLIANG TIAN received his B.S. in Computing from Macao Polytechnic University, Macao, China, in 2018. He is currently a Ph.D. student in Computer Science at Washington University in St. Louis. Since 2025, he has been a Research Assistant at Washington University in St. Louis. His research interests include quantum computing and networking.



RAJ JAIN (Life Fellow, IEEE) received a B.E. in electrical engineering from APS University, India, and an M.E. from the Indian Institute of Science. He obtained his Ph.D. in Applied Maths (computer science) from Harvard University. He is the Barbara J. and Jerome R. Cox, Jr., Professor at Washington University in St. Louis. Dr. Jain co-founded Nayna Networks, Inc., and has held significant roles in academia and industry. He has received the 2018 James B. Eads Award and the 2017 ACM SIGCOMM Life-Time Achievement Award. He is known as one of the most cited authors in computer science and has authored "The Art of Computer Systems Performance Analysis." He is a Fellow of IEEE, ACM, and AAAS.



RAMANA KOMPELLA received a B.Tech. from the Indian Institute of Technology, Bombay (1999), an M.S. from Stanford University (2001), and a Ph.D. from the University of California, San Diego (2007), all in computer science and engineering. He serves as a Distinguished Engineer and the Head of Research in Cisco's Emerging Tech and Incubation group, leading university research collaborations. His research focuses on data centers, cloud performance optimization, and router algorithmics. Dr. Kompella has been an ACM member since 2007, received the NSF CAREER Award in 2011, and received Best Paper awards at conferences such as ACM SOCC.



REZA NEJABATI is the Head of Quantum Research and Quantum Lab at Cisco, focused on advancing quantum networking. Previously, he was Chair Professor of Networks and Head of the High-Performance Network Group at the University of Bristol, UK. He is also a visiting professor and Cisco Chair at Curtin University's Cisco Center for Intent-Based Networking in Australia. Reza established globally recognized research in autonomous and quantum networks, co-founded Zeetta Networks Ltd, and received the IEEE Charles Kao Award in 2016 for contributions to 5G, smart cities, quantum communication, and future Internet experimentation.



ENEET KAUR is a Quantum Research Scientist at Cisco. Her research focuses on quantum networking and quantum key distribution. She received her Ph.D. in Physics from Louisiana State University and conducted postdoctoral research at the University of Waterloo.



AIMAN ERBAD received the B.Sc. degree in computer engineering from the University of Washington, Seattle, in 2004, the Master of Computer Science degree in embedded systems and robotics from the University of Essex, U.K., in 2005, and the Ph.D. degree in computer science from the University of British Columbia, Canada, in 2012. He is an Associate Professor and the ICT Division Head of the College of Science and Engineering, Hamad Bin Khalifa University. He published more than 190 papers in reputable international conferences and journals. His research interests span cloud computing, quantum networks, edge intelligence, Internet of Things, and private and secure networks. He received the 2020 Best Research Paper Award from Computer Communications, the IWCMC 2019 Best Paper Award, and the IEEE CCWC 2017 Best Paper Award. He is the General Chair of ISNCC 2023. He also served as the Program Chair of IWCMC 2022 and IWCMC 2019, as the Publicity Chair of ACM MoVid 2015, as the Local Arrangement Chair of NOSSDAV 2011, and as a Technical Program Committee Member in various IEEE and ACM international conferences (GlobeCom, ICC, NOSSDAV, MMSys, ACMMM, IC2E, and ICNC). He is a Senior Member of ACM.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Quantum Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TQE.2026.3744686



MOHAMED ABDALLAH was born in Giza, Egypt. He received the B.Sc. degree with honors from Cairo University, Giza, Egypt, in 1996, and the M.Sc. and Ph.D. degrees in electrical engineering from the University of Maryland at College Park, College Park, MD, USA, in 2001 and 2006, respectively. He joined Cairo University in 2006, where he holds the position of Associate Professor in the Electronics and Electrical Communication Department. He is currently an Associate Research Scientist at Texas A\&M University at Qatar, Doha, Qatar. His current research interests include the design and performance of physical layer algorithms for cognitive networks, cellular heterogeneous networks, sensor networks, smart grids, visible light and free-space optical communication systems, and reconfigurable smart antenna systems.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/