frontiers | Frontiers in Computer Science

TYPE Hypothesis and Theory
PUBLISHED 16 July 2025
DOI 10.3389/fcomp.2025.1520903

OPEN ACCESS

EDITED BY
Sebastian Maneth,
University of Bremen, Germany

REVIEWED BY
Sabina Rossi,
Ca' Foscari University of Venice, Italy
Koffka Khan,
The University of the West Indies St. Augustine, Trinidad and Tobago

*CORRESPONDENCE
Edoardo Giusto
egiusto@ieee.org

RECEIVED 31 October 2024
ACCEPTED 11 June 2025
PUBLISHED 16 July 2025

CITATION
Giusto E, Núñez-Corrales S, Smith KN, Cao P, Younis E, Rech P, Vella F, Baheri B, Cilardo A, Montrucchio B, Jiang W, Xu S, Dasgupta S, Iyer RK and Humble TS (2025) Dependable classical-quantum computing systems engineering. *Front. Comput. Sci.* 7:1520903. doi: 10.3389/fcomp.2025.1520903

Dependable classical-quantum computing systems engineering

Edoardo Giusto[1]*, Santiago Núñez-Corrales[2], Kaitlin N. Smith[3], Phuong Cao[2,4], Ed Younis[5], Paolo Rech[6], Flavio Vella[7], Betis Baheri[8], Alessandro Cilardo[1], Bartolomeo Montrucchio[9], Weiwen Jiang[10], Shuai Xu[11], Samudra Dasgupta[12], Ravishankar K. Iyer[4] and Travis S. Humble[12]

[1]Department of Electrical Engineering and Information Technology, University of Naples Federico II, Naples, Italy, [2]National Center for Supercomputing Applications, University of Illinois Urbana-Champaign, Urbana, IL, United States, [3]Department of Computer Science, Northwestern University, Evanston, IL, United States, [4]University of Illinois Urbana-Champaign, Champaign, IL, United States, [5]Computer Science Department, Lawrence Berkeley National Laboratory, Berkeley, CA, United States, [6]Department of Industrial Engineering, University of Trento, Trento, Italy, [7]Department of Information Science and Engineering, University of Trento, Trento, Italy, [8]Department of Computer Science, Kent State University, Kent, OH, United States, [9]Department of Control and Computer Engineering, Politecnico di Torino, Torino, Italy, [10]Department of Electrical and Computer Engineering, George Mason University, Fairfax, VA, United States, [11]Department of Computer and Data Sciences, Case Western Reserve University, Cleveland, OH, United States, [12]Quantum Science Center, Oak Ridge National Laboratory, Oak Ridge, TN, United States

Increasing evidence suggests quantum computing (QC) complements traditional High-Performance Computing (HPC) by leveraging its unique capabilities, leading to the emergence of a new, hybrid paradigm, QHPC. However, this integration introduces new challenges, with dependability—defined by reproducibility, resiliency, and security and privacy—emerging as a central concern for building trustworthy systems that provide an advantage to the users. This paper proposes a framework for dependable QHPC system design, organized around these three pillars. We identify integration challenges, anticipate roadblocks, and highlight productive synergies across QC, HPC, cloud platforms, and network security. Drawing from both classical computing principles and quantum-specific insights, we present a roadmap for co-design that supports robust hybrid architectures. Our approach offers concrete metrics for assessing dependability, provides design guidance for engineers working at the QC-HPC interface, and surfaces new engineering questions around complexity, scale, and fault tolerance. Ultimately, designing for dependability is key to realizing practical, scalable QHPC systems and accelerating the broader quantum ecosystem capable of translating quantum promises into actual application delivery.

KEYWORDS

hybrid classical-quantum systems, HPC, quantum computing, dependability, reliability, resiliency, security, reproducibility

1 Introduction

Exploiting the expected potential benefits of QC in scientific and engineering applications will require its integration into HPC infrastructure, already the backbone of large-scale computing (Alexeev et al., 2024). This scenario ushers QHPC as a new computational paradigm (Britt and Humble, 2017) that involves the efficient and reliable incorporation of Quantum Processing Units (QPUs) into standard HPC

workflows (Saurabh et al., 2023; Matsuura and Mattson, 2022; Tang and Martonosi, 2024; Beck et al., 2024; Brown et al., 2024). Scaling up emerging Noisy Intermediate-Scale Quantum (NISQ) devices and integrating them with HPC infrastructure poses significant engineering challenges. This classical-quantum integration has triggered the exploration of multiple hardware (Britt and Humble, 2017) and software (Saurabh et al., 2023) pathways starting from a model of QPUs as remotely accessible computational accelerators, akin to how Graphics Processing Units (GPUs) being an integral part of HPC architectures (Cui et al., 2025). Recent advances highlight the integration of QPUs as HPC accelerators (McCaskey et al., 2018, 2020), the conceptualization of HPC-QPU enablers (Humble et al., 2021; Saurabh et al., 2023), the development of quantum kernels for scientific applications (Matsuura and Mattson, 2022), and progress in hybrid classical-quantum algorithms tailored to specific domains such as quantum chemistry and material science (Robledo-Moreno et al., 2024; Alexeev et al., 2024).

In this article, we highlight the need for dependable QHPC systems engineering by describing how it provides value from quantum hardware design to domain-specific applications, and how aspects of dependability usually found in HPC infrastructure will apply to quantum platforms as their integration into classical resources deepens. We do so by identifying how *reproducibility*, *resiliency*, and *security & privacy* (Figure 1) fit the new QHPC paradigm. Ultimately, dependability involves the use of systems that can be trusted. The complexity derived from the intersection across pillars is non-trivial in emerging QHPC environments, and calls for an all-encompassing solution from a high-level view that permeates across all elements of the stack (Figure 2).

Dependability in HPC systems and classical computing more generally starts at the hardware substrate. Modern computing is built upwards from computing elements (transistors) whose behavior is that of a bistable system with very low error rates ($\propto 10^{-18}$–10^{-24}). This means, in practice, that despite the occurrence of fluctuations at the hardware level, the likelihood of faults translating into errors is relatively low and can be handled gracefully. QPUs, on the contrary, are best described as metastable systems—systems whose stability requires constant, active maintenance—with large error rates in comparison ($\propto 10^{-4}$–10^{-7}). From the start, operating and integrating QPUs into HPC systems bears a larger difficulty just by virtue of the real-time control problem involved under various sources of environment perturbation. As an engineering consequence, the complexity of QHPC stacks is necessarily greater than that of pure HPC systems by virtue of Ashby's law of requisite variety (Boisot and McKelvey, 2011). More complexity creates opportunities for interactions—and, by extension, classes of faults—across layers of the stack for which traditional dependability techniques and theory are insufficient to anticipate and mitigate undesirable whole-system states.

We envision for the purposes of our discussion a high-level hybrid architecture structured in three layers: the *Scientific Workload Layer*, the *System Management Layer*, and the *Hardware Layer*. The *Scientific Workload Layer* serves as the uppermost stratum, encompassing HPC applications, quantum algorithms, and hybrid workflows using classical and quantum resources. The intermediate *System Management Layer* orchestrates the distribution of computational tasks, implementing sophisticated load balancing mechanisms and resource management protocols. This layer provides the first set of opportunities to optimize the utilization of both classical and quantum resources while maintaining system efficiency. The foundational *Hardware Layer* is split into two components: classical hardware (i.e., CPUs, GPUs, TPUs, and hierarchical memory systems) and quantum hardware (i.e., QPUs and their associated control systems). Compilation and transpilation of the programs derived by the workload specified above occur here, verifying that the produced set of instructions actually implement the desired task, and the execution of such tasks on heterogeneous hybrid quantum-classical hardware.

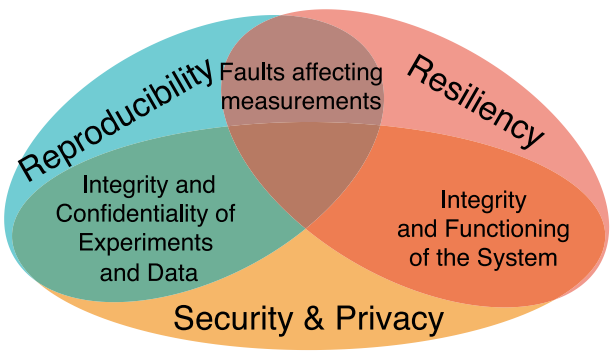

FIGURE 1
The Venn diagram highlights the intersection of reproducibility, resiliency and security in creating dependable computing systems.

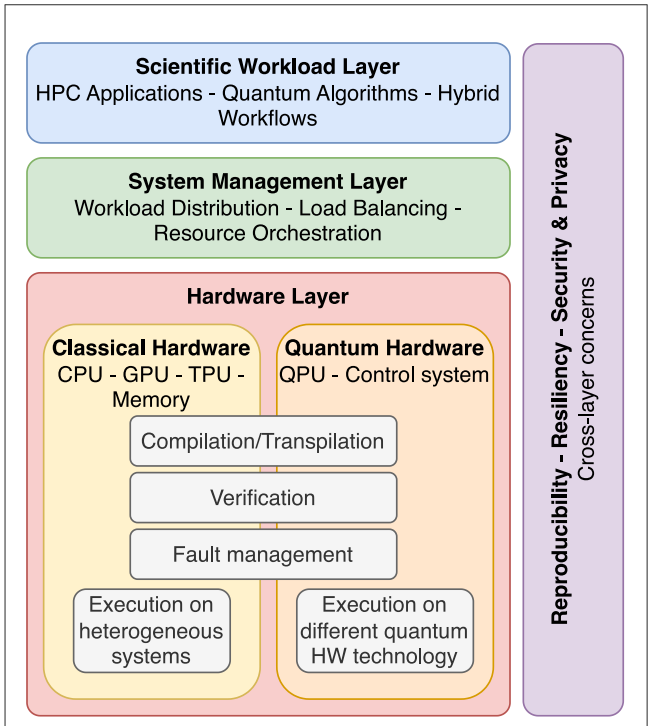

FIGURE 2
QHPC high-level architecture and related cross-layer dependability concerns.

Concerns about dependability across all three pillars arise promptly throughout these layers. Reproducibility is involved when managing the execution of a scientific calculation workflow in such a way that the produced output is consistent and repeatable between different executions and systems; this is challenging when the behavior of quantum resources varies rapidly and non-deterministically. Resiliency distinguishes between correct vs. incorrect states of execution across hardware and software across multiple abstraction levels in classical and quantum portions of the stack; the relatively small number and scale of available quantum systems limits our information of their faults and failures. Associated concerns include the compilation, verification, and execution of code on heterogeneous systems. Security and Privacy determine trustworthiness from the user's perspective, which eventually needs to be maintained across the entire spectrum of the system components and processes to ensure that the aforementioned workflow is not hampered by an external agent. In short, QHPC systems must protect intellectual property (IP) and user privacy with similar requirements as those put on purely classical cyberinfrastructure; we have gained only limited understanding on the extent of possible vulnerabilities introduced by quantum computing besides post-quantum cryptography.

In consequence, our manuscript intends to contribute toward achieving the following objectives:

- Introduction of a framework for dependability at the intersection of HPC and QC.
- Precise characterization of the concept of dependability within this framework.
- Identification of roadblocks and challenges to achieve dependability in this hybrid domain.
- Definition of a set of engineering principles of dependability in this context.
- Impact of fault-tolerant quantum systems for dependability in QHPC.
- Introduction of an R&D roadmap for dependable HPC and QC integration.

The rest of the paper is organized as follows. Sections 2, 3, 4 address each of the pillars of dependability, while Section 5 brings them all together, laying out the challenge of scaling up dependable QHPC systems and discussing high-level implications and research needs for this integration. Finally, Section 6 concludes the paper.

2 Reproducibility

Reproducibility is a fundamental principle in science and the ability to communicate and transfer knowledge. In the presence of noise, quantum computers may become unreliable, in a formal sense, as their behaviors are no longer strictly predictable. Statistical notions of operations and their outcomes become necessary for describing such noisy quantum computers. Moreover, results from quantum programs executed on noisy devices raise concerns about how to statistically quantify reproducibility in the presence of noise and errors (Dasgupta and Humble, 2022b,a; Hu et al., 2023; Senapati et al., 2023).

2.1 Accuracy vs. stability

The *accuracy* of a QC system represents the agreement between the observed and expected results of the QC program. For conventional analysis, binary outcomes from measuring the quantum register after execution of a quantum program (currently expressed as a quantum circuit) constitute the objects of interest. Histograms drawn from these outcomes characterize the *stability* of the computation, which varies unpredictably due to non-stationary stochastic processes and their statistical observables (Dasgupta and Humble, 2023). Whereas *accuracy* is a measure of error in the calculation itself, *stability* quantifies the fluctuations in such observations with respect to time. Large and unpredictable fluctuations in these quantum processor observables are a fundamental concern for the reproducibility of results.

These uncertainties arise since practical efforts to build quantum computers introduce unexpected sources of noise through various types of imperfections (Gao et al., 2021; Wintersperger et al., 2023). In consequence, imperfections cause quantum devices to depart from idealized computing behavior. Quantum computers exhibit drifting noise landscapes as a result (Dasgupta and Humble, 2024). Example noise sources include spontaneous decay of qubits, leakage from the computational subspace, undesired external coupling due to spurious charge and magnetic fields, and inter-qubit cross-talk from capacitive coupling. Similarly, noise in the control system arises from imperfections in the fundamental gate operations, e.g., in superconducting qubits, distortion and drift in microwave pulses often lead to errors. Externally, a multitude of mechanisms are put in place to isolate the quantum device and stave off decoherence as it interacts with its environment (e.g., ultra-low temperatures, ultra-low vacuum). Thus, anticipating how HPC environments (acoustic noise, vibrations, etc.) will impinge on QC device operation becomes inescapable.

Remark 1: Nonstationary noise in contemporary quantum devices presents a challenge for computational reproducibility that impacts the production of trustworthy results at the level of scientific applications.

2.2 Measuring quantum reproducibility

To measure reproducibility in quantum computing systems, we extend the analysis of stability to discrete distributions with binary outcomes $\{f_b\}$ from quantum stochastic processes. To achieve this, we choose the Hellinger distance among various statistical distance measures as it extends to both discrete and continuous distributions, thus satisfying the requirements of a distance metric for comparison. For distributions $f(b)$ and $g(b)$, the Hellinger distance $H(f,g) \in [0,1]$ (Lindsay, 1994) is defined as

$$H(f,g) = \sqrt{1 - B(f,q)} \tag{1}$$

with Bhattacharyya coefficient

$$B(f,g) = \sum_{b} \sqrt{f(b)g(b)} \tag{2}$$

The results are δ-reproducible with tolerance ϵ when

$$\Pr(H(f,g) \leq \epsilon) \geq 1 - \delta \tag{3}$$

For quantum programs, f and g are computed for a given quantum circuit using multiple shots. As an example, the minimum sample size $L_{\min}$ required for reproducibility for the Bernstein-Vazirani algorithm can be shown to be a non-linear function of the confidence level $1 - \delta$ and the accuracy threshold ϵ:

$$L_{\min} = z_{\delta}^2 \frac{p_r^{-2} - 1}{p_r^{-2}(1-\epsilon)^2 - 1} \tag{4}$$

where z represents the standard normal variable (with mean 0 and variance 1), z_{δ} denotes the particular point where $\Pr(z \geq z_{\delta}) = 1 - \delta$, and p_r signifies the probability of successfully identifying the secret string r using the Bernstein-Vazirani algorithm in the presence of noise. A pressing concern is what values of the tolerances, ϵ and δ, are sufficient for real-world applications of dependable QC.

3 Resiliency

Resiliency characterizes the ability of a system to maintain a desired state given a range of perturbations, making it trusted and effective in performing, while being capable of providing detection and graceful degradation of function and performance (Goerger et al., 2014). In computer systems, hardware and software components are expected to undertake this responsibility. At the software level, executable code must accurately describe a computation based on a specification that aligns with the architectural constraints of the quantum processor. However, this machine-friendly abstraction—a *model* of the hardware—must be generated automatically, without needing the quantum application programmer to be aware of the requirements and capabilities of the hardware. This conversion between representations is done via *compilation* and *transpilation* of a quantum workload from logical circuits to low-level control instructions, often electromagnetic pulses to be applied to the physical qubits. As the compilation pipeline becomes more robust, *verified* translation should be prioritized to ensure the *trustworthiness* of compiled outputs.

At the hardware level, *faults* arising in one or more physical units of the system should ideally not disrupt the execution flow or alter the reproducibility of the results; more realistically, the execution of a quantum program should be accompanied by information about the impact of faults. Faults, error detection, and subsequent recovery in classical HPC has been well-studied, with several management techniques proposed (Canal et al., 2020). Our understanding of faults in quantum technology is still maturing. When we discuss faults in quantum systems, we highlight that what is considered a fault can range in impact. For example, faults can range from temporary fluctuations in error rates due to external radiation or cosmic rays (Martinis, 2021) to permanent degradation of two-qubit gate fidelity because of a frequency collision defect set from an imperfect fabrication process (Smith et al., 2022a).

In this section, we briefly review compilation, transpilation, and the necessary verifications in quantum program generation, as well as faults in classical and quantum systems, and discuss both classical and quantum techniques to counteract such faults, analyzing whether applying classical dependability techniques to the quantum domain is possible.

3.1 Compilation and verification

Computation starts with formulating a problem and synthesizing a formal specification in the form of code into an intermediate representation (IR) for reasoning, manipulation, and optimization on an algorithmic level. In QC, the quantum circuit model that utilizes quantum registers, quantum operations, classical registers, and basic classical logic is often used as a quantum application IR (Cross et al., 2022). Specifications written using the quantum circuit model are often referred to as quantum circuits or programs. After the initial quantum program or circuit generation, compilation transforms the technology-independent IR into a technology-compatible form. This involves converting the high-level algorithmic building blocks to a low-level device-specific instructions. This conversion process must preserve the semantics of the original quantum program while making sure that the new version of the circuit agrees with the QPU constraints, such as qubit-qubit connectivity and native gate set, so that the circuit can be physically executed (Smith and Thornton, 2019).

Compilation is a complex process that often occurs in multiple steps or passes. In addition to transforming logical operations to physically-realized instructions, programs are often optimized according to hardware characteristics, coherence times, and noise properties to improve quantum resource utilization and increase outcome fidelity (Campbell et al., 2023). At the lowest level, optimization procedures can also include instruction scheduling (Smith et al., 2022b; Ravi et al., 2022), insertion of corrective gate operations (Das et al., 2021), and creation of custom gates via quantum optimal control (Shi et al., 2019).

In tandem, quantum circuit verification is critical to ensure correctness and reliability. This entails addressing both the classical and quantum aspects derived from the probabilistic nature of quantum computation and noisy hardware. Currently, this is achieved through a combination of circuit equivalence-based classical formal techniques and transpilation passes verification (Wille and Burgholzer, 2022; Younis et al., 2021). Modern compilers verify the circuit transformations applied during the transpilation and lowering stages.

The compilation and verification of quantum programs must then be put in the context of hybrid QHPC environments. This brings a plethora of new concerns, such as workload balancing optimization for specific and heterogeneous architectures (CPU+GPU+TPU+QPU), enabling parallel execution on such heterogeneous architectures, performance tuning, profiling, and program optimization. Formal verification must then be overlaid

across this complexity to guarantee that requirements are fulfilled throughout the entire composite system.

Additionally, cross-platform program compilation must ensure reproducible behavior across potentially different hybrid technologies and architectures. This requires the compiler itself to make informed decisions about the best methods to implement the programming model and to coordinate between the quantum and conventional computing resources. A paramount concern is that the hybrid program must yield the same result regardless of which architecture it runs on, as numerical accuracy is a non-negotiable requirement. The compiler will enforce this requirement through choices made by the programmer and computer architect on how to compile and implement hybrid instructions.

Early results have already demonstrated the dramatic impact of quantum architecture on quantum compilation (Linke et al., 2017), and we are now beginning to understand how choices in the accompanying HPC architecture will also impact performance (Ang et al., 2024). Moreover, as near-term devices are not yet stable or stationary with respect to their noise, modifications to the underlying error management methods, including Quantum Error Correction (QEC) techniques, will require re-compilation on timescales associated with noise fluctuations (Sivak et al., 2023). Recent work shows how the logical qubit mapping problem accommodates for a realistic and faulty lattices of physical qubits (Lin et al., 2024; Yin et al., 2024).

Remark 2: Compilation processes must be adapted to heterogeneous QHPC architectures. The execution of a quantum program must be verified across the entire stack, from high-level specifications to sequences of pulses for physical devices.

3.2 Fault management

Classical-quantum programs may fail to run for reasons well beyond compilation and verification errors. More generally, system or application failure can be caused by faults/errors at different levels of the system hierarchy (e.g., hardware, operating system, communication layer, middleware, or application). Their cascading effects thus call for a system perspective: we need to anticipate and engineer mechanisms that produce resilient computing systems as a result. Similar to reproducibility, we need metrics to quantify the resiliency of an QHPC system.

3.2.1 Classical computing fault management

Initially, fault tolerance has been defined as the ability to avoid catastrophic disruptions, e.g., a total system failure, in the presence of errors (Avizienis et al., 2004). With the advent of cloud computing systems, precise service-level agreements (SLA) emerged as the contracts that a system must adhere to keep customers' workloads running. In this context, a formal definition of fault tolerance is the ability of a system to maintain critical fault-tolerant operating metrics (Pham et al., 2014, 2012; Iyer et al., 2024) within the SLA as follows.

These metrics are: (i) the *Mean Time To Failure (MTTF)* or *Mean Time To Error (MTTE)*, (ii) the *Mean Time Between Failures (MTBF)*, (iii) the *Mean Time To Catastrophic Failure (MTTCF)*, (iv) the *fault latency*, (v) the *error latency*, and (vi) the *Mean Time to Repair (MTTR)*. These metrics circumscribe the extent of a *fault cycle* (Figure 3) and determine the *dimensions* of resiliency: *reliability*, *maintainability*, *availability*, *safety*.

To clarify the terminology used here, *faults* are underlying causes or defects, which may be dormant and may not immediately impact the system operation; *errors* are the manifestations of faults, occurring when the faults become active, representing a deviation from the correct state and may potentially propagate to cause system *failure*. The relationship follows a sequential chain:

$$\text{Fault} \rightarrow \text{Error} \rightarrow \text{(potential) Failure}$$

As an example, particle impacts constitute a major source of faults in classical computing systems. These are a naturally occurring byproduct of cosmic-ray decay which corrupts stored values and the executed operations in both classical and quantum devices. In a CMOS transistor, ionizing particles generate electron-hole pairs, releasing and depositing charge (Baumann, 2005). A sufficiently large deposited charge forces a transistor state to flip with three possible consequences. There may not be an effect on program output if the fault is *masked*, or the corrupted data is not used. In another case, known as *silent data corruption* (SDC), the program yields incorrect results, yet continues to run. Finally, a bit flip may trigger a *detected unrecoverable error* (DUE) in which the program crashes or the device is forced to reboot.

Figure 4 summarizes widespread techniques available in the classical domain to handle faults across the system hierarchy (Iyer et al., 2024). In the following, we evaluate the applicability of classical techniques to the quantum domain, but it is first important to mention two theorems in the quantum domain that may hinder such applicability: the *Threshold Theorem* and the *No-Cloning Theorem*.

The *Threshold Theorem* proves the existence of a critical error rate below which we can build fault-tolerant quantum computers (FTQC) (Paler and Devitt, 2015; Nielsen and Chuang, 2010). This means that, even with imperfectly realized qubits and gates, we can perform arbitrarily long computations with high accuracy (Roffe, 2019). The solution is implemented using quantum error correction codes (QECC), which encode a single logical qubit using multiple physical qubits to achieve robustness, incurring an overhead to detect and fix errors as they happen and cascade through the computation. QECC remains an active research area (Campbell, 2024).

Next, the *No-Cloning Theorem* states the impossibility to create a perfect copy of an arbitrary unknown quantum state (which can be a superposition of $|0\rangle$ and $|1\rangle$) to another qubit without disturbing the original state (Wootters and Zurek, 1982). Apart from requiring quantum operations to be reversible, it places a hard limit in our ability to store and retrieve purely quantum states in the context of projective measurements. This means, in fact, that several assumptions behind classical techniques used to gain reliability become unavailable for quantum portions of QHPC systems, and different routes may be required. More research is required to understand the impact of no-cloning in this regard.

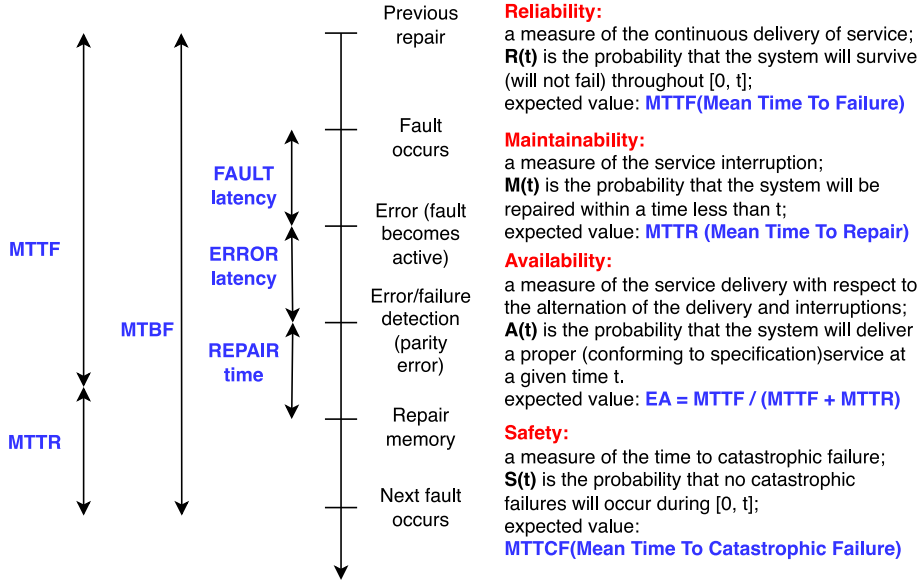

FIGURE 3
Resiliency metrics and dimensions in relation to the fault cycle of a classical system.

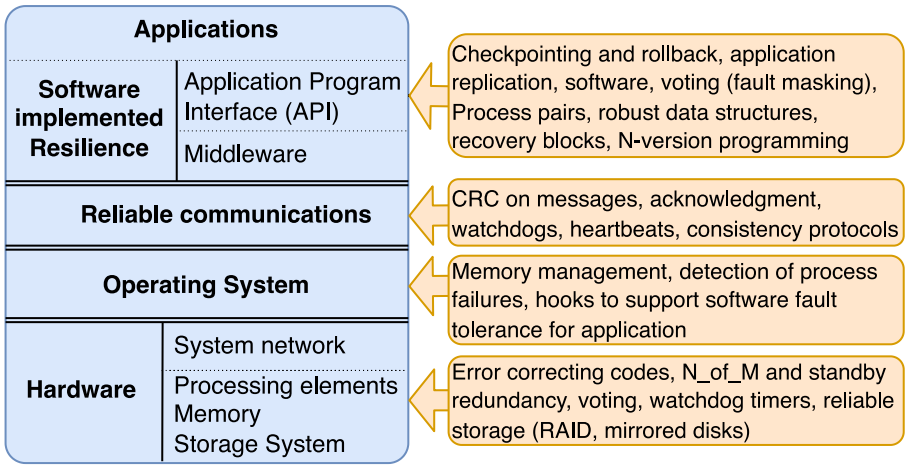

FIGURE 4
Classical techniques available at each level of a system.

3.2.2 Applicability of classical techniques to the quantum domain

In light of these theorems, we now report the main techniques, evaluating their applicability to the quantum domain. *Replication* consists of running the same task on multiple processing units simultaneously. Due to the limitations imposed by the No-Cloning Theorem, what is currently done is the subsequent repeated execution of the circuit under consideration to statistically estimate the resulting state vector's probabilities. Full replication would entail running circuit ensembles in parallel on a set of identical devices, which is cost-prohibitive in existing quantum hardware, which sits below utility thresholds for practical applications. Instead, *job resubmission* entails re-executing failed jobs, potentially on different resources, to improve the chances of successful completion. This can be done in QC if an error detection technique is implemented to identify sporadic external errors such as those induced by radiation, but not for errors due to inherent decoherence.

Remark 3: The Threshold and No-cloning theorems place fundamental limitations on our ability to apply classical techniques to quantum computing systems. The landscape of consequences originating in these limitations remains mostly unexplored for the dependability of QHPC systems. Classical-quantum systems will force the community to re-think resiliency across the stack.

3.2.3 Quantum computing fault management

In QC devices, the impact of particles alters the state of qubit(s) by forcing them into decoherence, dictated by physics of light-matter and matter-matter interactions. For instance, a fault mechanism in superconducting devices involves the generation of electron-hole pairs in the silicon substrate of the quantum chip, which in turn break Cooper pairs in the Josephson junction forming quasiparticles, that rapidly give rise to long-lasting phonons responsible for spreading the energy across the lattice of the quantum computer's substrate and interconnections (Vepsäläinen et al., 2020; Wilen et al., 2021). While in a classical transistor the state is temporary reversed only if the deposited charge by the particle is higher than a threshold, even a single Cooper pair break is sufficient to disturb the quantum equilibrium of a qubit, thus modifying the logic status.

Field experiments performed by Google AI on a 25 qubits array showed radiation-induced faults every tens of seconds (McEwen et al., 2022). The reported error rate is several orders of magnitude higher than the one of modern CMOS technology. As a reference, the whole Titan supercomputer (composed of 14,000 nodes) had an error rate in the order of one error every few hours (Tiwari et al., 2015). As of now, there is no well-established taxonomy for fault in the quantum domain as it is for the classical domain. A *Quantum Vulnerability Factor* has been recently proposed to quantify the effect of faults at different levels of abstraction in the execution of quantum circuits (Oliveira et al., 2024, 2022). At the physical level, *hardware improvements* such as quasiparticle traps (Martinis, 2021) and superconducting gap engineering (McEwen et al., 2024) are continuously proposed to mitigate internal and external noise factors.

Remark 4: Faults add combinatorial randomness on top of intrinsic noise, changing the output probability distribution of the computation non-deterministically. These are likely to have larger impact on qubits than on classical CMOS transistors, and disrupt qubit behavior for a longer time.

3.2.4 Toward fault tolerant quantum computing

The development of techniques addressing unique failure concerns for QC has gradually acquired interest. These techniques are organized in three categories, namely quantum error suppression (QES), quantum error mitigation (QEM), and quantum error correction (QEC).

QES integrates resiliency directly at the physical level, preventing faults from arising through precise control of the quantum system. This is done optimizing pulse engineering to minimize errors during gate operations (Mundada et al., 2023). The primary techniques include dynamic decoupling (DD) and optimal control methods, which have shown significant promise in maintaining quantum coherence (Wang and Liu, 2011; Caneva et al., 2011). From a systems perspective, implementing QES presents challenges in generating and timing control pulses with sufficient precision. While hardware constraints continue to bound achievable control fidelity, machine learning-assisted pulse optimization could provide a foundation for more sophisticated error handling techniques (Krenn et al., 2023).

Remark 5: QES integrates resiliency at the hardware level through precise control of quantum systems, but faces significant challenges in pulse engineering precision. Machine learning-assisted optimization offers a promising path forward.

QEM builds upon error suppression techniques by employing probabilistic methods to further reduce computational errors. This approach has gained prominence due to its minimal qubit overhead and compatibility with near-term quantum devices (Temme et al., 2017). Among QEM methods, *zero-noise extrapolation* and *probabilistic error cancellation* have emerged as leading techniques for improving the accuracy of quantum computations without additional quantum resources. Current methods must balance the tradeoff between sampling overhead and error reduction, particularly for deep circuits where exponential overhead becomes prohibitive.

Remark 6: QEM provides a practical approach for near-term devices with minimal qubit overhead, but faces an exponential tradeoff between sampling overhead and error reduction for deep circuits.

QEC employs redundancy across multiple physical qubits to detect and correct errors. This is not without challenges due to resource overhead and complex operations (Roffe, 2019), as well as the presence of correlated faults (Martinis, 2021). Among QEC codes, the *surface code*, particularly the family of *rotated surface codes*, has become a leading candidate due to its robustness against bit-flip and phase-flip errors and its scalability for larger quantum systems. In a system integration perspective, a major challenge in implementing QEC is managing the latency of the error correction process. Quantum processors generate vast amounts of data—∼1 TB per second—that need to be decoded in real-time (Barber et al., 2025).

Decoders must process this data within stringent time constraints, typically $<1ms$, to ensure synchronization with quantum operations (Battistel et al., 2023). Current decoders, such as Union-Find and Minimum Weight Perfect Matching (MWPM), are computationally efficient but struggle to balance speed and accuracy. More advanced methods, such as Tensor Network (TN) contractions and Belief Propagation (BP) hybrids, offer better error-correcting performance but are computationally intensive, exacerbating the challenge of handling large data volumes. The Collision Clustering (CC) (Barber et al., 2023) decoder provides a promising path forward by combining the scalability of Union-Find with optimized memory usage, this results in almost linear decoding algorithms. Recent developments, however, suggest that the code distance necessary for fault tolerance can be reduced (Acharya et al., 2024). While managing the massive data output and meeting real-time decoding requirements remains a challenge, the reduction in code distance offers a pathway to more resource-efficient and scalable quantum computers. Several open-ended questions remain around QEC in the presence of faults, such as how to design practical error correction implementations that bridge the gap between the heterogeneous

noise landscape of physical resources and application fidelity and runtime latency requirements.

Remark 7: Effective QEC codes depend on overcoming the challenges of latency, data processing, and scalability. Decoding methods like MWPM, Union-Find, TN, and BP each have their strengths, but none individually meet all the demands of large-scale, real-time QEC. Co-designing such methods with efficient programmable low-latency system architectures is still needed.

4 Security and privacy

The complex interfaces between traditional HPC, QC, and emerging applications introduce various new security issues. First, classical systems face threats from quantum-driven adversaries—a prime example being how traditional cryptography based on large integer factorization, such as RSA, can be broken by quantum computers running Shor's algorithm (Gidney and Ekerå, 2021). Second, the security of quantum systems themselves presents challenges, particularly regarding information confidentiality, data integrity, and ensuring availability of quantum services, which are by-large accessed via the Quantum Cloud (Ravi et al., 2021) and Jupyter Notebooks (Cao, 2024), to end users. Third, maintaining user data privacy and regulatory compliance becomes crucial when using QHPC to process sensitive research and health data, such as Controlled Unclassified Information (CUI) and Health Insurance Portability and Accountability Act (HIPAA) data.

We further lay out research directions in this hybrid domain for each of these scenarios as follows.

4.1 Quantum-driven adversaries against quantum and classical systems

Attackers have been leveraging accelerators such as GPUs to crack passwords, and will use QPUs to upend traditional cryptographic systems, e.g., RSA, as the National Institute of Standards and Technology urged a quantum-resistant cryptography transition by 2030. Initial evidence shows that as GPUs become more powerful, artificial intelligence (AI)-driven malware can learn when and how to launch stealth data-stealing campaigns to minimize their footprints (Chung et al., 2023). Our hypothesis is that QPUs will be leveraged to launch unforeseen attacks well beyond exploiting improved algorithmic complexity for cryptographic problems into finding vulnerabilities across problems that require multiparty coordination.

Despite cross-stack hardened security (Powell, 2018), we lack theory, experience, and real-world measurements on how large-scale quantum computing systems can enable malicious attacks across the QHPC stack. In this context, the problem of standardizing post-quantum cryptography (PQC), such as lattice-based cryptography, and measuring the adoption rate (Sowa et al., 2024) in different domains (Carroll et al., 2024) will provide critical feedback to policy makers such as NIST.

Remark 8: Modern quantum-driven attacks will leverage the power of QPUs to
1) mask the presence of attacks as natural faults,
2) adapt attack traces to minimize their footprint, and
3) learn when, how, and where to launch attacks in a QC stack to maximize damage.

4.2 Attacks at the surface of QHPC integration

The initial adoption of QHPC will start at research computing centers and national labs, which are ripe for attacks due to their open-science environment. New attack surfaces will emerge as physicists, computer scientists, and engineers join forces to integrate HPC and QC, as all sides will make different assumptions on QHPC interfaces (Maurya et al., 2024).

Classical HPC-targeting attacks (Yang et al., 2024) such as federated authentication in open-science research (Basney et al., 2020), credential and key stuffing (Wu et al., 2020) have been studied. However, to observe new QHPC-targeting attacks, network security monitors, such as Zeek (Paxson, 1999; Cao et al., 2019), must be adapted to the hybrid workload and data being transferred through the QHPC interface to gain visibility on adversaries operating on encrypted traffic (Piet et al., 2023).

As Quantum Key Distribution (QKD) systems and control mechanisms mature, the classical systems become the weakest link in QHPC integration and need careful security engineering and monitoring across the stack. This landscape is populated mainly with *unknown unknowns*, and is ripe for scientific and technical discovery.

Remark 9: Preemptive attack detection (Cao et al., 2015) at the surface of QHPC is critical to ensure secure integration, which requires
1) inventing a new instrument that monitors quantum network data,
2) distributing security monitoring instruments across networks for early detection of attacks, and
3) defining checkpointing and remedy mechanisms to respond to attacks.

4.3 Privacy concerns in integrated QHPC platforms

Privacy-Preserving QC is critical as QC services, presumed to be untrusted, will be adopted in domains where data privacy and intellectual property are an inherent requirement, such as drug discovery, financial optimization, and machine learning for health applications.

4.3.1 Pure cryptography-based privacy-preserving approach

Confidential QC approaches adopt classical cryptographic schemes for privacy-preserving computing, i.e. Fully Homomorphic Encryption (FHE) and Multi-Party

Computation (MPC). In particular, Blind Quantum Computing (BQC) (Fitzsimons, 2017) partially delegates quantum computation to a remote server without disclosing the computation itself. More recent proposals for BQC relaxed this requirement (Huang et al., 2017), but still need an assumption of multiple non-colluding servers, which may not be realistic in practice. Quantum Homomorphic Encryption (QHE) (Ouyang et al., 2018), on the other hand, enables the quantum computer to work on encrypted quantum data and produce an encrypted outcome, without accessing data, similar to its classical counterpart, but it remains impractical as it involves exponential computational overheads.

4.3.2 A trusted quantum computing base (TQCB) for privacy-preserving

Classical Trusted Execution Environments (TEEs) rely on hardware that isolates computation within a chip, offering security. Quantum computers, with their distributed data and components, pose challenges to redefining the Trusted Computing Base (Trochatos et al., 2024) as researchers are exploring ways to establish natively quantum TEEs for protecting sensitive data. Active areas of exploration in the space of quantum TEEs include hardware fingerprinting schemes for QPUs in the quantum cloud (Allen et al., 2021; Smith et al., 2023; Wu et al., 2024), quantum physically unclonable functions (Arapinis et al., 2021; Doosti et al., 2021; Phalak et al., 2021; Smith and Gokhale, 2023), and the concept of a quantum antivirus (Deshpande et al., 2022).

Remark 10: Standardizing data privacy and security following the Findability, Accessibility, Interoperability, and Reusability (FAIR) principles (Wilkinson et al., 2016), and more importantly making specifications *executable*, is critically needed to enable:
1) secure-by-construction data access and sharing protocols,
2) formal verification of the entire quantum computing stack and synthesizing corresponding correct implementations, and
3) security data lake and testbed (Cao et al., 2024) for evaluating unforeseen attack scenarios, such as data leaks or compromising the root of trust.

We expect the research community to engage with NIST and other agencies to identify needs and later define standards (Aydeger et al., 2024). This type of work acquires relevance against the backdrop of increasing export controls regulations (Bauer and Pandya, 2024) and ongoing geopolitical transformations (Liman and Weber, 2023; Der Derian and Rollo, 2024).

5 Discussion

To become a dependable technology capable of delivering tangible value to users with performance-critical scientific applications, quantum hardware platforms must adapt to the operational demands and environments of existing HPC systems. The challenge of achieving successful QHPC integration remains largely unresolved, as the fundamental differences between classical and quantum technologies, coupled with their disparate levels of maturity, introduce significant complexity. Our work—as well as cited works from others—suggests the necessity of a new research domain at the intersection between classical HPC and quantum computing. The title of this article, *Dependable Classical-Quantum Computing Systems Engineering*, deliberately qualifies the nature of the endeavor as one drawing heavily from scientific principles to articulate usable systems, not to advance the science of devices *per se*. While experimental quantum testbeds have been instrumental for advancing quantum hardware research, we suggest that dedicated QHPC testbeds are necessary for conducting a holistic and rigorous analysis of dependability. The key dependability principles summarized in Table 1 provide a structured framework for this analysis, covering all pillars.

TABLE 1 Dependability principles for hybrid classical-quantum computing systems.

No.	Principle	Description
1	Hardware-aware compilation	Leverage real-time hardware status (classical and quantum) to drive compilation and load balancing decisions, ensuring programs map efficiently onto heterogeneous QHPC architectures
2	Measurable reproducibility	Quantify reproducibility via statistical tolerances ϵ (accuracy) and δ (confidence)
3	Quantitative resilience metrics	Define and track dependability metrics over the fault cycle, adapted to QC device noise and fault models
4	Verified cross-platform consistency	Enforce formal verification of compilation/transpilation transformations and ensure identical hybrid workflow outcomes regardless of underlying QHPC hardware
5	Security & privacy by design	Integrate post-quantum cryptography, blind/fully homomorphic quantum computing, and trusted execution constructs to protect IP, data integrity, and user privacy across the QHPC stack
6	Open co-design and standards	Develop open hybrid OS and compilers with standardized IRs, testbeds, and interfaces to drive transparent hardware/software co-design

The three pillars of dependability are deeply intertwined into QHPC systems: the more applications and research users hinge on these, the more sophisticated the ability of the system to satisfy essential guarantees must be. Application users, beginning from high-level scientific workload specifications, primarily focus on solution delivery—effectively, computing as a service—with minimal concern for the underlying stack implementation. Nevertheless, these users demand performance, reproducibility, resiliency, and security in their scientific endeavors. Building dependable classical-quantum hardware and software components multiplies the complexity of HPC systems.

Quantum computing is moving closer to building utility-scale, fault-tolerant quantum systems capable of solving complex, real-world problems. From a system perspective, current solutions to this computational demand often require application-specific integrated circuit (ASIC) designs to meet real-time performance, limiting scalability and programmability. ASIC-based approaches are efficient but tightly coupled to the specific quantum system's requirements, making them less adaptable to evolving quantum hardware. Understanding the impact of FTQC

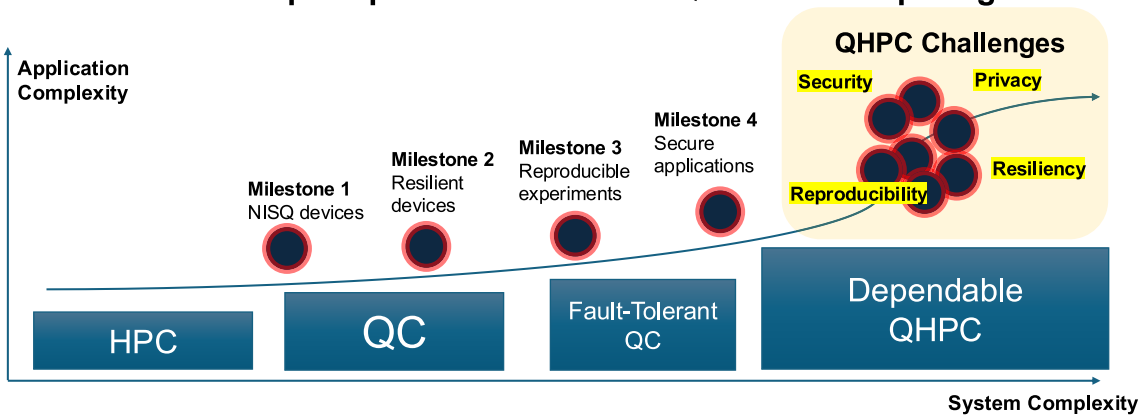

FIGURE 5
Roadmap for dependable QHPC.

systems across dependability is increasingly becoming a highly relevant problem.

We advocate for full engagement across the HPC, networking, and QC communities, pursuing three primary objectives. First, to delineate specific problem domains where current solutions remain intuitive rather than systematic—our observations suggest the need for interdisciplinary collaboration to address multiple uncertainty sources. Second, to establish development milestones for dependable QHPC systems that leverage ongoing QC community achievements. These milestones should align with QHPC hardware testbed advancement and accelerate full-system hardware-software co-design. A successful selection of milestones will be characterized by a healthy balance between new avenues of exploration and effective ways to discard unproductive research directions. Third and final, to cultivate broader dialogue leading to a comprehensive roadmap—similar to that presented in Figure 5—for implementing a sustainable, long-term QHPC integration program.

5.1 Future perspectives

Technical progress across multiple directions necessitates transparent access to both software and hardware systems. Contemporary HPC systems are predominantly built on top of open-source stacks, allowing complete code inspection and comprehensive hardware interrogation at all meaningful levels. An equivalent ecosystem for classical-quantum computing has yet to emerge. In this context, the creation of an operating system (OS) for hybrid quantum-classical systems is highly desirable, and a natural consequence of integration in the long run. However, tackling this challenge early provides multiple advantages today. The development of a natively quantum-classical OS would substantiate the co-design principle by exploiting current status of execution and hardware devices themselves to drive processes such as compilation and load balancing in order to improve efficiency and maximizing usage of the system. We need to investigate what managing quantum resources alongside classical ones means, and devise technologies to achieve it. The creation of such an OS will prompt the definition of standards, specifying the way information is transmitted and processed. This can happen top-down, from the high level workload definition to the sequence of pulses to apply to the qubits, and bottom up, making upper layers aware of the status of the hardware for informed decision making (i.e. hardware aware compilation, load balancing, etc.). In this context, reaching consensus on the adoption of IR (or a set of IRs) at different levels of the stack is important, yet guided by principles operating at a higher-order. In tandem, we advocate for the creation of an open compiler for hybrid systems exploiting such sets of standard IRs and information flow streams, ensuring continued accessibility and fostering the development of new tools for achieving dependable QHPC. Succinctly, developing a quantum OS will force the community to design systems that serve users in need of advance computing resources and advance via opportunistic refinement, not experiments whose scope is limited to quantum information science and engineering.

One of the key components of future quantum OSs will most certainly be Artificial Intelligence (AI). AI models can flexibly adapt compilation policies to changing configurations (hardware availability and status, task queues, etc.) and constraints (classical, AI, and quantum workloads) (Teranishi et al., 2025). AI agents will be in charge of orchestrating workloads across the heterogeneity of hardware platforms with the overarching goal of optimizing resource utilization and increasing throughput while reducing energy consumption (Chen et al., 2024). Moreover, the synergy of AI and blockchain technologies has strong potential to support emerging applications. AI can enhance blockchain performance in key areas such as consensus, scalability, and anomaly detection, enabling smarter, more autonomous decision-making at runtime (Ressi et al., 2024). Conversely, blockchain's inherent transparency and immutability may support trusted audit trails of quantum-classical workloads, which is critical for reproducibility and explainability in high-stakes computations. Looking ahead, quantum-enhanced blockchains represent another frontier, offering the promise of quantum-secure consensus mechanisms and advanced quantum cryptographic primitives like quantum money and collapsing hash functions (Edwards et al., 2020). These tools could be instrumental in designing

tamper-proof control flows and verifiable distributed coordination across hybrid systems. As quantum computing matures, integrating these elements into the core fabric of a QHPC OS will be essential but for resilience and long-term trust.

6 Conclusion

This article shows that, despite experience gathered across several decades of HPC practice, the introduction of QC brings new and interesting challenges. Ensuring the dependability of hybrid systems is paramount in order to leverage such computational power for scientific computing applications. This paper describes the three pillars of dependability—reproducibility, resiliency, and security & privacy—in the context of QHPC integration. Reproducibility is affected by quantum noise, while classical approaches in the resiliency domain may have fundamental shortcomings if applied to the quantum realm. Security threats are amplified due to the intricate integration of such heterogeneous components. Overcoming the dependability challenges is crucial to enable reliable, high-performance scientific applications that leverage heterogeneous resources.

Our work is call to arms for experts across the HPC, QC, cybersecurity, and any other relevant communities to come together and address these issues in a unified front. We argue that developing a combined co-design approach, supported by open testbeds, can pave the way toward dependable, robust QHPC platforms that can deliver on the revolutionary promise of this emerging paradigm.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

EG: Writing – original draft, Writing – review & editing. SN-C: Writing – original draft, Writing – review & editing. KS: Writing – original draft, Writing – review & editing. PC: Writing – original draft, Writing – review & editing. EY: Writing – original draft, Writing – review & editing. PR: Writing – original draft, Writing – review & editing. FV: Writing – original draft, Writing – review & editing. BB: Writing – original draft, Writing – review & editing. AC: Writing – original draft, Writing – review & editing. BM: Writing – original draft. WJ: Writing – original draft. SX: Writing – original draft. SD: Writing – original draft. RI: Writing – original draft. TH: Writing – original draft, Writing – review & editing.

Funding

The author(s) declare that financial support was received for the research and/or publication of this article. This work has been partially supported by the the Spoke 9 of the ICSC National Research Centre for High Performance Computing, Big Data and Quantum Computing. This work has been partially funded by the National Center for Supercomputing Applications, Illinois Computes, New Frontier Initiative, and IBM-Illinois Discovery Accelerator Institute at the University of Illinois Urbana-Champaign; Trusted CI: The NSF Cybersecurity Center of Excellence; U.S. National Science Foundation grants #1547249, #1535070, #1935966, #2029049, #2319190, #2430244. We thank ACCESS and DeltaAI program for providing compute infrastructure and storage. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the funders.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declare that Gen AI was used in the creation of this manuscript. The authors declare that AI tools were used to help edit the manuscript, summarizing, and restructuring text.

Publisher's note

References

Acharya, R., Abanin, D. A., Aghababaie-Beni, L., Aleiner, I., Andersen, T. I., Ansmann, M., et al. (2024). Quantum error correction below the surface code threshold. *Nature* 638, 920–926. doi: 10.1038/s41586-024-08449-y

Alexeev, Y., Amsler, M., Barroca, M. A., Bassini, S., Battelle, T., Camps, D., et al. (2024). Quantum-centric supercomputing for materials science: a perspective on challenges and future directions. *Future Gener. Comput. Syst.* 160, 666–710. doi: 10.1016/j.future.2024.04.060

Allen, M., Shuwen, D., and Jakub, S. (2021). "Short paper: device-and locality-specific fingerprinting of shared nisq quantum computers," in *Workshop on Hardware and Architectural Support for Security and Privacy* (New York, NY: ACM), 1–6. doi: 10.1145/3505253.3505261

Ang, J., Carini, G., Chen, Y., Chuang, I., Demarco, M., Economou, S., et al. (2024). Arquin: Architectures for multinode superconducting quantum computers. *ACM Trans. Quantum Comput.* 5, 1–59. doi: 10.1145/3674151

Arapinis, M., Delavar, M., Doosti, M., and Kashefi, E. (2021). Quantum physical unclonable functions: possibilities and impossibilities. *Quantum* 5:475. doi: 10.22331/q-2021-06-15-475

Avizienis, A., Laprie, J.-C., Randell, B., and Landwehr, C. (2004). Basic concepts and taxonomy of dependable and secure computing. *IEEE Trans. Dependable Secure Comput.* 1, 11–33. doi: 10.1109/TDSC.2004.2

Aydeger, A., Zeydan, E., Awaneesh, K., Hemachandra, K. T., Mishra, S., Liyanage, M., et al. (2024). "Towards a quantum-resilient future: strategies for transitioning to post-quantum cryptography," in *15th International Conference on Network of the Future (NoF)* (Castelldefels: IEEE). doi: 10.1109/NoF62948.2024.10741441

Barber, B., Barnes, K. M., Bialas, T., Buğdaycı, O., Campbell, E. T., Gillespie, N. I., et al. (2023). A real-time, scalable, fast and highly resource efficient decoder for a quantum computer. *arXiv* [Preprint]. arXiv:2309.05558. doi: 10.48550/arXiv:2309.05558

Barber, B., Barnes, K. M., Bialas, T., Buğdaycı, O., Campbell, E. T., Gillespie, N. I., et al. (2025). A real-time, scalable, fast and resource-efficient decoder for a quantum computer. *Nat. Electron.* 1–8.

Basney, J., Cao, P., and Fleury, T. (2020). "Investigating root causes of authentication failures using a saml and oidc observatory," in *2020 IEEE 6th International Conference on Dependability in Sensor, Cloud and Big Data Systems and Application (DependSys)* (Nadi: IEEE), 119–126. doi: 10.1109/DependSys51298.2020.00026

Battistel, F., Chamberland, C., Johar, K., Overwater, R. W. J., Sebastiano, F., Skoric, L., et al. (2023). Real-time decoding for fault-tolerant quantum computing: progress, challenges and outlook. *Nano Futures* 7:032003. doi: 10.1088/2399-1984/aceba6

Bauer, M., and Pandya, D. (2024). *Time to Rethink Export Controls for Strengthened US-EU Cooperation and Global Trade Rules (No. 07/2024)*. Brussels: ECIPE Policy Brief; European Centre for International Political Economy (ECIPE).

Baumann, R. (2005). Soft errors in advanced computer systems. *IEEE Des. Test Comput.* 22, 258–266. doi: 10.1109/MDT.2005.69

Beck, T., Baroni, A., Bennink, R., Buchs, G., Pérez, E. A. C., Eisenbach, M., et al. (2024). Integrating quantum computing resources into scientific HPC ecosystems. *Future Gener. Comput. Syst.* 161, 11–25. doi: 10.1016/j.future.2024.06.058

Boisot, M., and McKelvey, B. (2011). "Complexity and organization-environment relations: revisiting Ashby's law of requisite variety," in *Sage Handbook of Complexity and Management*, eds. P. Allen, S. Maguire, and B. McKelvey (London: SAGE), 279–298.

Britt, K. A., and Humble, T. S. (2017). High-performance computing with quantum processing units. *ACM J. Emerg. Technol. in Comput. Syst.* 13, 1–13. doi: 10.1145/3007651

Brown, K., Chong, F., Smith, K. N., Conte, T., Adams, A., Dalvi, A., et al. (2024). 5 year update to the next steps in quantum computing. *arXiv preprint* arXiv:2403.08780.

Campbell, C., Chong, F. T., Dahl, D., Frederick, P., Goiporia, P., Gokhale, P., et al. (2023). "Superstaq: deep optimization of quantum programs," in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE), Volume 1* (Bellevue, WA: IEEE), 1020–1032. doi: 10.1109/QCE57702.2023.00116

Campbell, E. (2024). A series of fast-paced advances in quantum error correction. *Nat. Rev. Phys.* 6, 160–161. doi: 10.1038/s42254-024-00706-3

Canal, R., Hernandez, C., Tornero, R., Cilardo, A., Massari, G., Reghenzani, F., et al. (2020). Predictive reliability and fault management in exascale systems: state of the art and perspectives. *ACM Comput. Surv.* 53, 1–32. doi: 10.1145/3403956

Caneva, T., Calarco, T., and Montangero, S. (2011). Chopped random-basis quantum optimization. *Phys. Rev. A–At. Mol. Opt. Phys.* 84:022326. doi: 10.1103/PhysRevA.84.022326

Cao, P. (2024). "Jupyter notebook attacks taxonomy: ransomware, data exfiltration, and security misconfiguration," in *SC24-W: Workshops of the International Conference for High Performance Computing, Networking, Storage and Analysis* (IEEE), 750–754. doi: 10.1109/SCW63240.2024.00106

Cao, P., Badger, E., Kalbarczyk, Z., Iyer, R., and Slagell, A. (2015). "Preemptive intrusion detection: theoretical framework and real-world measurements," in *Proceedings of the 2015 Symposium and Bootcamp on the Science of Security* (New York, NY: ACM), 1–12. doi: 10.1145/2746194.2746199

Cao, P., Kalbarczyk, Z., and Iyer, R. K. (2024). "Security testbed for preempting attacks against supercomputing infrastructure," in *SC24-W: Workshops of the International Conference for High Performance Computing, Networking, Storage and Analysis* (Atlanta, GA: IEEE), 1781–1788. doi: 10.1109/SCW63240.2024.00223

Cao, P. M., Wu, Y., Banerjee, S. S., Azoff, J., Withers, A., Kalbarczyk, Z. T., et al. (2019). "CAUDIT: continuous auditing of SSH servers to mitigate Brute-Force attacks," in *16th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19)*, 667682.

Carroll, T. E., Moran-Schmoker, A. M., and Redington, L. M. (2024). "Exploring the adoption challenges of post-quantum cryptography in EV charging infrastructure," in *Technical report, Pacific Northwest National Laboratory (PNNL)* (Richland, WA). doi: 10.2172/2337525

Chen, D., Youssef, A., Pendse, R., Schleife, A., Clark, B. K., Hamann, H., et al. (2024). Transforming the hybrid cloud for emerging ai workloads. *arXiv* [Preprint]. arXiv:2411.13239. doi: 10.48550/arXiv:2411.13239

Chung, K., Cao, P., Kalbarczyk, Z. T., and Iyer, R. K. (2023). "stealthml: Data-driven malware for stealthy data exfiltration," in *2023 IEEE International Conference on Cyber Security and Resilience (CSR)* (Venice: IEEE), 16–21. doi: 10.1109/CSR57506.2023.10224946

Cross, A., Javadi-Abhari, A., Alexander, T., De Beaudrap, N., Bishop, L. S., Heidel, J. M., et al. (2022). Openqasm 3: a broader and deeper quantum assembly language. *ACM Trans. Quantum Comput.* 3, 1–50. doi: 10.1145/3505636

Cui, S., Patke, A., Chen, Z., Ranjan, A., Nguyen, H., Cao, P., et al. (2025). Characterizing gpu resilience and impact on ai/hpc systems. *arXiv* [Preprint]. arXiv:2503.11901. doi: 10.48550/arXiv.2503.11901

Das, P., Tannu, S., Dangwal, S., and Qureshi, M. (2021). "Adapt: mitigating idling errors in qubits via adaptive dynamical decoupling," in MICRO-*54: 54th Annual IEEE/ACM International Symposium on Microarchitecture* (New York, NY: ACM), 950–962. doi: 10.1145/3466752.3480059

Dasgupta, S., and Humble, T. (2024). Impact of unreliable devices on stability of quantum computations. *ACM Transac. Quantum Comput.* 5, 1–23. doi: 10.1145/3682071

Dasgupta, S., and Humble, T. S. (2022a). "Assessing the stability of noisy quantum computation," in *Quantum Communications and Quantum Imaging XX, Volume 12238* (SPIE), 44–49. doi: 10.1117/12.2631809

Dasgupta, S., and Humble, T. S. (2022b). Characterizing the reproducibility of noisy quantum circuits. *Entropy* 24:244. doi: 10.3390/e24020244

Dasgupta, S., and Humble, T. S. (2023). Reliability of noisy quantum computing devices. *arXiv* [Preprint]. arXiv:2307.06833. doi: 10.48550/arXiv.2307.06833

Der Derian, J., and Rollo, S. (2024). "quantum 3.0": what will it mean for war, peace, and world order? *Glob. Perspect.* 5:93888. doi: 10.1525/gp.2024.93888

Deshpande, S., Xu, C., Trochatos, T., Ding, Y., and Szefer, J. (2022). "Towards an antivirus for quantum computers," in *2022 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)* (McLean, VA: IEEE), 37–40. doi: 10.1109/HOST54066.2022.9840181

Doosti, M., Kumar, N., Delavar, M., and Kashefi, E. (2021). Client-server identification protocols with quantum puf. *ACM Trans. Quantum Comput.* 2, 1–40. doi: 10.1145/3484197

Edwards, M., Mashatan, A., and Ghose, S. (2020). A review of quantum and hybrid quantum/classical blockchain protocols. *Quantum Inf. Process.* 19, 1–22. doi: 10.1007/s11128-020-02672-y

Fitzsimons, J. F. (2017). Private quantum computation: an introduction to blind quantum computing and related protocols. *NPJ Quantum Inf.* 3:23. doi: 10.1038/s41534-017-0025-3

Gao, Y. Y., Rol, M. A., Touzard, S., and Wang, C. (2021). Practical guide for building superconducting quantum devices. *PRX Quantum* 2:040202. doi: 10.1103/PRXQuantum.2.040202

Gidney, C., and Ekerå, M. (2021). How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits. *Quantum* 5:433. doi: 10.22331/q-2021-04-15-433

Goerger, S. R., Madni, A. M., and Eslinger, O. J. (2014). Engineered resilient systems: A DOD perspective. *Procedia Comput. Sci.* 28, 865–872. doi: 10.1016/j.procs.2014.03.103

Hu, Z., Wolle, R., Tian, M., Guan, Q., Humble, T., Jiang, W., et al. (2023). "Toward consistent high-fidelity quantum learning on unstable devices via efficient in-situ calibration," in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE), Volume 1* (Bellevue, WA: IEEE), 848–858. doi: 10.1109/QCE57702.2023.00099

Huang, H.-L., Zhao, Q., Ma, X., Liu, C., Su, Z.-E., Wang, X.-L., et al. (2017). Experimental blind quantum computing for a classical client. *Phys. Rev. Lett.* 119:050503. doi: 10.1103/PhysRevLett.119.050503

Humble, T. S., McCaskey, A., Lyakh, D. I., Gowrishankar, M., Frisch, A., Monz, T., et al. (2021). Quantum computers for high-performance computing. *IEEE Micro* 41, 15–23. doi: 10.1109/MM.2021.3099140

Iyer, R. K., Kalbarczyk, Z. T., and Nakka, N. M. (2024). *Dependable Computing: Design and Assessment*. Hoboken, NJ: John Wiley & Sons. doi: 10.1002/9781119743453

Krenn, M., Landgraf, J., Foesel, T., and Marquardt, F. (2023). Artificial intelligence and machine learning for quantum technologies. *Phys. Rev. A* 107:010101. doi: 10.1103/PhysRevA.107.010101

Liman, A., and Weber, K. (2023). Quantum computing: bridging the national security-digital sovereignty divide. *Eur. J. Risk Regul.* 14, 476–483. doi: 10.1017/err.2023.44

Lin, S. F., Viszlai, J., Smith, K. N., Ravi, G. S., Yuan, C., Chong, F. T., et al. (2024). "Codesign of quantum error-correcting codes and modular chiplets in the presence of defects," in *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume* 2 (New York, NY: ACM), 216–231. doi: 10.1145/3620665.3640362

Lindsay, B. G. (1994). Efficiency versus robustness: the case for minimum hellinger distance and related methods. *Ann Stat.* 22, 1081–1114. doi: 10.1214/aos/1176325512

Linke, N. M., Maslov, D., Roetteler, M., Debnath, S., Figgatt, C., Landsman, K. A., et al. (2017). Experimental comparison of two quantum computing architectures. *Proc. Natl. Acad. Sci.* 114, 3305–3310. doi: 10.1073/pnas.1618020114

Martinis, J. M. (2021). Saving superconducting quantum processors from decay and correlated errors generated by gamma and cosmic rays. *NPJ Quantum Inf.* 7:90. doi: 10.1038/s41534-021-00431-0

Matsuura, A., and Mattson, T. G. (2022). Introducing the quantum research kernels: lessons from classical parallel computing. *arXiv* [Preprint]. arXiv:2211.00844. doi: 10.4855/arXiv.2211.00844

Maurya, S., Mude, C. N., Lienhard, B., and Tannu, S. (2024). Understanding side-channel vulnerabilities in superconducting qubit readout architectures. *arXiv* [Preprint]. arXiv:2405.08962. doi: 10.1109/QCE60285.2024.00138

McCaskey, A. J., Dumitrescu, E. F., Liakh, D., and Chen, M. Feng, W.-c., Humble, T. S. (2018). A language and hardware independent approach to quantum-classical computing. *SoftwareX* 7, 245–254. doi: 10.1016/j.softx.2018.07.007

McCaskey, A. J., Lyakh, D. I., Dumitrescu, E. F., Powers, S. S., and Humble, T. S. (2020). Xacc: a system-level software infrastructure for heterogeneous quantum-classical computing. *Quantum Sci. Technol.* 5:024002. doi: 10.1088/2058-9565/ab6bf6

McEwen, M., Faoro, L., Arya, K., Dunsworth, A., Huang, T., Kim, S., et al. (2022). Resolving catastrophic error bursts from cosmic raysrin large arrays of superconducting qubits. *Nat. Phys.* 18, 107–111. doi: 10.1038/s41567-021-01432-8

McEwen, M., Miao, K. C., Atalaya, J., Bilmes, A., Crook, A., Bovaird, J., et al. (2024). Resisting high-energy impact events through gap engineering in superconducting qubit arrays. *arXiv* [Preprint]. arXiv:2402.15644. doi: 10.48550/arXiv.2402.15644

Mundada, P. S., Barbosa, A., Maity, S., Wang, Y., Merkh, T., Stace, T., et al. (2023). Experimental benchmarking of an automated deterministic error-suppression workflow for quantum algorithms. *Phys. Rev. Appl.* 20:024034. doi: 10.1103/PhysRevApplied.20.024034

Nielsen, M. A., and Chuang, I. L. (2010). *Quantum Computation and Quantum Information*. Cambridge: Cambridge university press.

Oliveira, D., Giusto, E., Baheri, B., Guan, Q. Montrucchio, B., and Rech, P. (2024). A systematic methodology to compute the quantum vulnerability factors for quantum circuits. *IEEE Trans. Depend. Secure Comput.* 21, 26312644. doi: 10.1109/TDSC.2023.3313934

Oliveira, D., Giusto, E., Dri, E., Casciola, N., Baheri, B., Guan, Q., et al. (2022). "QUFI: a quantum fault injector to measure the reliability of qubits and quantum circuits," in *2022 52nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)* (Baltimore, MD: IEEE), 137–149. doi: 10.1109/DSN53405.2022.00025

Ouyang, Y., Tan, S. H., and Fitzsimons, J. F. (2018). Quantum homomorphic encryption from quantum codes. *Phys. Rev. A.* 98:042334. doi: 10.1103/PhysRevA.98.042334

Paler, A., and Devitt, S. J. (2015). An introduction to fault-tolerant quantum computing. *arXiv* [Preprint]. arXiv:1508.03695. doi: 10.48550/arXiv.1508.03695

Paxson, V. (1999). Bro: a system for detecting network intruders in real-time. *Comput. Netw.* 31, 2435–2463. doi: 10.1016/S1389-1286(99)00112-7

Phalak, K., Ash-Saki, A., Alam, M., Topaloglu, R. O., and Ghosh, S. (2021). Quantum puf for security and trust in quantum computing. *IEEE J. Emerg. Sel. Top. Circuits Syst.* 11, 333–342. doi: 10.1109/JETCAS.2021.3077024

Pham, C., Cao, P., Kalbarczyk, Z., and Iyer, R. K. (2012). "Toward a high availability cloud: techniques and challenges," in *IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN 2012)* (Boston, MA: IEEE), 1–6. doi: 10.1109/DSNW.2012.6264687

Pham, C., Estrada, Z., Cao, P., Kalbarczyk, Z., and Iyer, R. K. (2014). "Reliability and security monitoring of virtual machines using hardware architectural invariants," in *2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks* (Atlanta, GA: IEEE), 13–24. doi: 10.1109/DSN.2014.19

Piet, J., Nwoji, D., and Paxson, V. (2023). "Ggfast: automating generation of flexible network traffic classifiers," in *Proceedings of the ACM SIGCOMM 2023 Conference* (New York, NY: ACM), 850–866. doi: 10.1145/3603269.3604840

Powell, W. A. (2018). "High-performance spaceflight computing (HPSC) project overview," in *Radiation Hardened Electronics Technology Conference (RHET) 2018, number GSFC-E-DAA-TN62651*.

Ravi, G. S., Smith, K. N., Gokhale, P., and Chong, F. T. (2021). "Quantum computing in the cloud: analyzing job and machine characteristics," in *2021 IEEE International Symposium on Workload Characterization (IISWC)* (Storrs, CT: IEEE), 39–50. doi: 10.1109/IISWC53511.2021.00015

Ravi, G. S., Smith, K. N., Gokhale, P., Mari, A., Earnest, N., Javadi-Abhari, A., et al. (2022). "VAQEM: a variational approach to quantum error mitigation," in *2022 IEEE International Symposium on High-Performance Computer Architecture (HPCA)* (Seoul: IEEE), 288–303. doi: 10.1109/HPCA53966.2022.00029

Ressi, D., Romanello, R., Piazza, C., and Rossi, S. (2024). AI-enhanced blockchain technology: a review of advancements and opportunities. *J. Netw. Comput. Appl.* 225:103858. doi: 10.1016/j.jnca.2024.103858

Robledo-Moreno, J., Motta, M., Haas, H., Javadi-Abhari, A., Jurcevic, P., Kirby, W., et al. (2024). Chemistry beyond exact solutions on a quantum-centric supercomputer. *arXiv* [Preprint]. arXiv:2405.05068. doi: 10.48550/arXiv.2405.05068

Roffe, J. (2019). Quantum error correction: an introductory guide. *Contemp. Phys.* 60, 226–245. doi: 10.1080/00107514.2019.1667078

Saurabh, N., Jha, S., and Luckow, A. (2023). "A conceptual architecture for a quantum-HPC middleware," in *2023 IEEE International Conference on Quantum Software (QSW)* (Chicago, IL: IEEE), 116–127. doi: 10.1109/QSW59989.2023.00023

Senapati, P., Wang, Z., Jiang, W., Humble, T. S., Fang, B., Xu, S., et al. (2023). "Towards redefining the reproducibility in quantum computing: a data analysis approach on NISQ devices," in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE), Volume 1* (Bellevue, WA: IEEE), 468–474. doi: 10.1109/QCE57702.2023.00060

Shi, Y., Leung, N., Gokhale, P., Rossi, Z., Schuster, D. I., Hoffmann, H., et al. (2019). "Optimized compilation of aggregated instructions for realistic quantum computers," in *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems* (New York, NY: ACM), 1031–1044. doi: 10.1145/3297858.3304018

Sivak, V., Eickbusch, A., Royer, B., Singh, S., Tsioutsios, I., Ganjam, S., et al. (2023). Real-time quantum error correction beyond break-even. *Nature* 616, 50–55. doi: 10.1038/s41586-023-05782-6

Smith, K. N., and Gokhale, P. (2023). Trustworthy quantum computation through quantum physical unclonable functions. *arXiv* [Preprint]. arXiv:2311.07094. doi: 10.48550/arXiv.2311.07094

Smith, K. N., Ravi, G. S., Baker, J. M., and Chong, F. T. (2022a). "Scaling superconducting quantum computers with chiplet architectures," in *2022 55th IEEE/ACM International Symposium on Microarchitecture (MICRO)* (Chicago, IL: IEEE), 1092–1109. doi: 10.1109/MICRO56248.2022.00078

Smith, K. N., Ravi, G. S., Murali, P., Baker, J. M., Earnest, N., Javadi-Cabhari, A., et al. (2022b). Timestitch: exploiting slack to mitigate decoherence in quantum circuits. *ACM Trans. Quantum Comput.* 4, 1–27. doi: 10.1145/3548778

Smith, K. N., and Thornton, M. A. (2019). "A quantum computational compiler and design tool for technology-specific targets," in *Proceedings of the 46th International Symposium on Computer Architecture* (New York, NY: ACM), 579–588. doi: 10.1145/3307650.3322262

Smith, K. N., Viszlai, J., Seifert, L. M., Baker, J. M., Szefer, J., Chong, F. T., et al. (2023). "Fast fingerprinting of cloud-based nisq quantum computers," in *2023 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)* (San Jose, CA: IEEE), 1–12. doi: 10.1109/HOST55118.2023.10133778

Sowa, J., Hoang, B., Yeluru, A., Qie, S., Nikolich, A., Iyer, R., et al. (2024). "Post-quantum cryptography (PQC) network instrument: measuring pqc adoption rates and identifying migration pathways," in *2024 IEEE International Conference on Quantum Computing and Engineering (QCE), volume 01* (Montreal, QC: IEEE), 1835–1846. doi: 10.1109/QCE60285.2024.00213

Tang, W., and Martonosi, M. (2024). Distributed quantum computing via integrating quantum and classical computing. *Computer* 57, 131–136. doi: 10.1109/MC.2024.3360569

Temme, K., Bravyi, S., and Gambetta, J. M. (2017). Error mitigation for short-depth quantum circuits. *Phys. Rev. Lett.* 119:180509. doi: 10.1103/PhysRevLett.119.180509

Teranishi, K., Menon, H., Godoy, W. F., Balaprakash, P., Bau, D., Ben-Nun, T., et al. (2025). Leveraging ai for productive and trustworthy hpc software: challenges and research directions. *arXiv* [Preprint]. arXiv:2505.08135. doi: 10.48550/arXiv.2505.08135

Tiwari, D., Gupta, S., Rogers, J., Maxwell, D., Rech, P., Vazhkudai, S., et al. (2015). "Understanding GPU errors on large-scale HPC systems and the implications for system design and operation," in *2015 IEEE 21st International Symposium on High Performance Computer Architecture (HPCA)* (Burlingame, CA: IEEE), 331–342. doi: 10.1109/HPCA.2015.7056044

Trochatos, T., Xu, C., Deshpande, S., Lu, Y., Ding, Y., Szefer, J., et al. (2024). "A quantum computer trusted execution environment," in *2024 IEEE International Symposium on High-Performance Computer Architecture (HPCA)* (Edinburgh: IEEE), 613–613. doi: 10.1109/HPCA57654.2024.00051

Vepsäläinen, A. P., Karamlou, A. H., Orrell, J. L., Dogra, A. S., Loer, B., Vasconcelos, F., et al. (2020). Impact of ionizing radiation on superconducting qubit coherence. *Nature* 584, 551–556. doi: 10.1038/s41586-020-2619-8

Wang, Z.-Y., and Liu, R.-B. (2011). Protection of quantum systems by nested dynamical decoupling. *Phys. Rev. A* 83:022306. doi: 10.1103/PhysRevA.83.022306

Wilen, C. D., Abdullah, S., Kurinsky, N. A., Stanford, C., Cardani, L., D'Imperio, G., et al. (2021). Correlated charge noise and relaxation errors in superconducting qubits. *Nature* 594, 369–373. doi: 10.1038/s41586-021-03557-5

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., et al. (2016). The fair guiding principles for scientific data management and stewardship. *Sci. Data* 3, 1–9. doi: 10.1038/sdata.2016.18

Wille, R., and Burgholzer, L. (2022). "Verification of quantum circuits," in *Handbook of Computer Architecture*, ed. A. Chattopadhyay (Cham: Springer), 1–28. doi: 10.1007/978-981-15-6401-7_43-1

Wintersperger, K., Dommert, F., Ehmer, T., Hoursanov, A., Klepsch, J., Mauerer, W., et al. (2023). Neutral atom quantum computing hardware: performance and end-user perspective. *EPJ Quantum Technol.* 10:32. doi: 10.1140/epjqt/s40507-023-00190-1

Wootters, W. K., and Zurek, W. H. (1982). A single quantum cannot be cloned. *Nature* 299, 802–803. doi: 10.1038/299802a0

Wu, J., Hu, T., and Li, Q. (2024). Detecting fraudulent services on quantum cloud platforms via dynamic fingerprinting. *arXiv* [Preprint]. arXiv:2408.11203. doi: 10.48550/arXiv.2408.11203

Wu, Y., Cao, P., Withers, A., Kalbarczyk, Z. T., and Iyer, R. K. (2020). "Poster: mining threat intelligence from billion-scale SSH brute-force attacks," in *Workshop on Decentralized IoT Systems and Security, in Network and Distributed System Security (NDSS) Symposium*.

Yang, L., Chen, Z., Wang, C., Zhang, Z., Booma, S., Cao, P., et al. (2024). "True attacks, attack attempts, or benign triggers? an empirical measurement of network alerts in a security operations center," in *33rd USENIX Security Symposium (USENIX Security 24)*, 15251542.

Yin, K., Zhang, H., Shi, Y., Humble, T., Li, A., Ding, Y., et al. (2024). Flexiscd: flexible surface code deformer for dynamic defects. *arXiv* [Preprint]. arXiv:2405.06941. doi: 10.48550/arXiv.2405.06941

Younis, E., Iancu, C. C., Lavrijsen, W., Davis, M., and Smith, E. (2021). *Berkeley Quantum Synthesis Toolkit (BQSKIT) v1*. Berkeley, CA: Technical report, Lawrence Berkeley National Laboratory (LBNL).