

Understanding Logical-Shift Error Propagation in Quanyvolutional Neural Networks

Original

Understanding Logical-Shift Error Propagation in Quanyvolutional Neural Networks / Vallero, Marzio; Dri, Emanuele; Giusto, Edoardo; Montrucchio, Bartolomeo; Rech, Paolo. - In: IEEE TRANSACTIONS ON QUANTUM ENGINEERING. - ISSN 2689-1808. - 5:(2024), pp. 1-15. [10.1109/TQE.2024.3372880]

Availability:

This version is available at: 11583/2986649 since: 2024-03-07T15:00:34Z

Publisher:

IEEE

Published

DOI:10.1109/TQE.2024.3372880

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

Received 1 December 2023; revised 24 January 2024; accepted 27 February 2024; date of publication 5 March 2024;
date of current version 12 April 2024.

Digital Object Identifier 10.1109/TQE.2024.3372880

Understanding Logical-Shift Error Propagation in Quanvolutional Neural Networks

MARZIO VALLERO¹ , EMANUELE DRI² ,
EDOARDO GIUSTO^{2,3}  (Member, IEEE), BARTOLOMEO MONTRUCCHIO² ,
AND PAOLO RECH¹ 

¹University of Trento, 38122 Trento, Italy

²Politecnico di Torino, 10129 Torino, Italy

³University of Naples Federico II, 80125 Napoli, Italy

Corresponding author: Marzio Vallero (e-mail: marzio.vallero@unitn.it).

This work was supported in part by the Italian Ministry for Department of Industrial Engineering, University and Research (MUR) through the Departments of Excellence 2023-27 program under Grant L.232/2016 and in part by the CINECA award under the ISCRA initiative, for the availability of high-performance computing resources and support within the Q-CORE project.

ABSTRACT Quantonvolutional neural networks (QNNs) have been successful in image classification, exploiting inherent quantum capabilities to improve performance of traditional convolution. Unfortunately, the qubit's reliability can be a significant issue for QNNs inference, since its logical state can be altered by both intrinsic noise and by the interaction with natural radiation. In this article, we aim at investigating the propagation of logical-shift errors (i.e., the unexpected modification of the qubit state) in QNNs. We propose a bottom-up evaluation reporting data from 13 322 547 200 logical-shift injections. We characterize the error propagation in the quantum circuit implementing a single convolution and then in various designs of the same QNN, varying the dataset and the network depth. We track the logical-shift error propagation through the qubits, channels, and subgrids, identifying the faults that are more likely to cause misclassifications. We found that up to 10% of the injections in the quantonvolutional layer cause misclassification and even logical-shifts of small magnitude can be sufficient to disturb the network functionality. Our detailed analysis shows that corruptions in the qubits' state that alter their probability amplitude are more critical than the ones altering their phase, that some object classes are more likely than others to be corrupted, that the criticality of subgrids depends on the dataset, and that the control qubits, once corrupted, are more likely to modify the QNN output than the target qubits.

INDEX TERMS Fault injection, fault tolerance, quantum computing (QC), quantum machine learning (QML), reliability evaluation.

I. INTRODUCTION

In recent years, quantum computing (QC) has undergone tantalizing improvements that could broaden the classical concept of computation as a whole. With the current widespread access to simulators and quantum devices over the cloud, researchers have been able to quickly expand QC's reach to fields such as finance [1], chemistry [2], biomechanics [3], machine learning [4], [5], and many others. Quantum algorithms are implemented by encoding the input data in quantum bits (qubits) and executing quantum circuits, which are sequences of operations on one or more qubits. The *quantum advantage* is achieved by exploiting the quantum properties of qubits, namely, *superposition* and *entanglement*.

Lately, the potentiality of QC has been successfully applied to reduce the inefficiencies associated with the execution of convolution in traditional computing systems. Hybrid quantum-classical machine learning models, called quantonvolutional neural networks (QNNs) [6], deliver promising speedups in terms of convergence and inference times over the classical convolutional neural networks (CNNs) while maintaining a very similar classification accuracy [7].

The most challenging obstacle preventing quantum technology from thriving is reliability. Superconducting qubits, which are the most widely used quantum devices (adopted by IBM and Google, among others), ideally need to be operated at a temperature close to *absolute zero* and shielded

from *all* external interference, which is unfortunately unachievable. As a consequence, retention and relaxation errors shorten significantly the computationally useful lifetime of a qubit, inducing a logical-shift error in the qubit state. On top of this, it has been proven, through both experiments and simulations, that quantum devices are extremely sensitive to natural ionizing radiation [8], [9], [10], [11], [12], [13], [14], [15], [16]. Rather than flipping a bit, as would happen in complementary metal-oxide semiconductor (CMOS) technology [17], intrinsic noise and external radiation both trigger the qubit(s), modifying the resulting qubit logic state. Since the logic state of the qubit is not binary, the fault induces a *rotation* in the qubit(s) state(s), i.e., a logical-shift error.

While quantum error correction (QEC) strategies have been developed for mitigating single-qubit noise effects, their overhead is unacceptable for current noisy intermediate-scale quantum (NISQ) machines. In addition, the transient, correlated, and stochastic nature of radiation-induced faults would in any case make QEC ineffective since multiple qubits would be affected by the charge deposited by the particle. Thus, the current and foreseeable quantum technology will still need to deal with logical-shift errors. The goal of our evaluation is to understand if and how these faults impact the execution of QNNs. Despite the fact that extensive research to understand and improve the reliability of traditional neural networks has been triggered already [18], [19], [20], studies about fault propagation in QNNs are still lacking.

To fill this research gap, in this article, we propose a detailed investigation of the behavior of the QNN model based on the hardware efficient ansatz for implementing the quantum convolution. The quantum circuit we target is the starting point of a large number of current (and future) QNN models [7], [21], [22], [23], [24], [25]. Then, we showcase a methodology to track fault propagation in QNNs by considering three different implementations of the very first such model ever designed [6]. To the best of the authors' knowledge, this is the first work addressing the reliability of QNNs to logical-level faults. Although QNNs are rapidly evolving and not yet employed in the field, it is by no means premature—but rather absolutely urgent—to consider their reliability. By promptly addressing the issue imposed by both intrinsic and extrinsic radiation-induced faults, we can immediately start to develop new reliability solutions, rather than patching it up only after its impact will become evident in the field. In this article, we inject more than 13 *billion* logical-shift faults in the quantum layer, adapting to QNNs an open-source fault-injector for quantum circuits (*QuFI*) [16]. We aim at filling the gap in the reliability evaluation of QNNs by investigating and understanding how faults in the quantum layer propagate in the network during inference and why they cause misclassifications. We aim at advancing the knowledge of QNNs reliability by the following:

- 1) detailing a methodology to evaluate, through fault injection, the reliability of QNNs to logical-shift faults;

- 2) studying the reliability profile of the qLayer, identifying the more critical qubit(s) and how logical-shift faults modify the layer output;
- 3) measuring the probability for a logical-shift fault to cause a misclassification in QNNs;
- 4) understanding the fault-impact dependence on the input image, the dataset, and the QNN design;
- 5) assessing how the corruption of different subgrids or portions of the feature maps impacts the QNN accuracy.

The rest of this article is organized as follows. Section II provides background and related works in the field of QC and QNNs, Section III outlines the design space exploration of our evaluation, Section IV describes the adopted experimental setup, and Section V presents and discusses the experimental results and their implications. Section VI highlights the impact of the proposed methodology. Finally, Section VII concludes this article.

II. BACKGROUND AND RELATED WORK

This section covers the fundamentals of QC, QNNs, quantum noise, and radiation-induced faults. We aim at providing the necessary information to describe the context in which the proposed work has been carried out.

A. QUANTUM COMPUTING

In the classical computation domain, the smallest unit of information is a binary digit, which can either encode a 1 or a 0. Instead, the quantum computation paradigm uses a two-state quantum mechanical system, called *qubit*, which can exploit quantum properties, such as *superposition* and *entanglement*. The former allows a qubit to exist in multiple different states at once, while the latter is capable of linking multiple qubits into a higher level object, which displays correlation patterns among all its elements. A *superposition* state is represented by the linear combination of the basis states $|1\rangle$ and $|0\rangle$ according to a pair of complex probability amplitudes

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle. \quad (1)$$

Such a general formulation for $|\Psi\rangle$ can be visualized on the Bloch sphere, as seen in Fig. 1 (upper), mapping the quantum state onto a vector in spherical coordinates, thus controlled by the polar (ϕ) and azimuthal (θ) angles.

Quantum algorithms are executed by means of *quantum circuits*, described as a temporal sequence of possibly simultaneous operations (quantum gates) applied to specific qubits. Gates can operate on single qubits, or on multiple qubits. The latter are usually composed of one or more *control* qubits that condition the execution of a certain operation on one or more *target* qubits.

Since QC is probabilistic by nature, the circuit execution is repeated multiple times. Instead of having a single output, the circuit provides a probability distribution of qubit collapses from multiple runs. At the time of writing, quantum devices

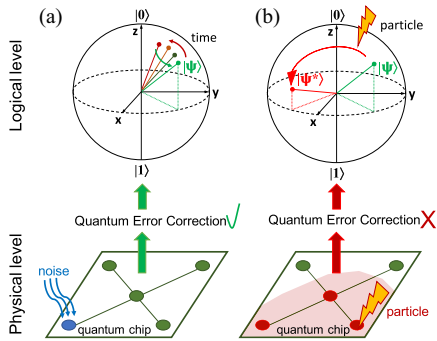


FIG. 1. Comparison of the effect of (a) noise and (b) particle impact at the physical and logical level of the qubit. The single logical qubit is implemented with several physical qubits (five in the picture) correlated by a QEC mechanism. Noise affecting one physical qubit, being well characterized, can be compensated for without affecting the logical qubit state. The charge deposited by the particle instead spreads across the whole physical substrate, jeopardizing QEC efficacy by generating a rotation of stochastic amplitude in the logical qubit's state, i.e., an error syndrome that cannot be corrected.

still belong to the NISQ era. These devices are capable of successfully executing only small algorithms, since technology development, in terms of control and insulation, has not yet reached the standards for fault tolerance. This also implies that some qubits can experience intrinsic noise that changes their state. Moreover, as detailed in Section II-B, despite the transient nature of radiation-induced faults, their persistence is orders of magnitude longer than the overall quantum circuit runtime [8]. Unfortunately, then, the repeated executions do not mitigate all possible sources of error.

Quantum circuits are defined at a logical, high abstraction level, only to be later *transpiled* into sequences of basic operations, the ones that can be directly carried out by architecture of each specific quantum device. The transpilation process involves mapping the quantum circuit onto a system of imperfect components, taking into account optimizations, noise reduction metrics, and heuristics [26].

B. QUANTUM NOISE AND RADIATION-INDUCED FAULTS

Logical qubits are exceptionally complex to implement and control on physical devices. Among the available technologies, we focus on the superconducting transmon qubit, since it is, by far, one of the most promising and widely adopted technologies.

To prevent the qubit's information from being corrupted, it must be completely isolated from the external environment, a task hardly achievable. Real qubits are characterized by two decoherence times, T_1 and T_2 . The former, called *spin-lattice coherence time*, refers to the natural energy decay time of an excited qubit in state $|1\rangle$ to the ground state $|0\rangle$. The latter, or *spin-relaxation time*, is the minimum interval before a qubit's state gets affected by external interference or by neighboring qubits. Fig. 1(a) shows a simplified visualization of how noise gradually modifies the qubit logic state.

Error handling techniques are continuously improved to increase T_1 and T_2 , in order to preserve quantum properties

for a longer time. QEC mechanisms, as depicted in Fig. 1, exploit hardware redundancy to ensure an accurate computation output even if the state of a qubit has degraded during the circuit execution. The quantum state of a single logical qubit is encoded into multiple physical qubits (as an example, five physical qubits in Fig. 1). Algorithms have to be adapted to act across multiple qubits and to apply the QEC. Currently, the most promising solution is the use of surface codes [27], which require at least 2^d physical qubits for each logical qubit, where d represents the *distance*, i.e., the number of errors that can be corrected. Unfortunately, the high cost of QEC makes it impractical for current NISQ machines. At the time of writing, most quantum device providers map logical qubits directly to physical ones, thus logical-shifts in qubits, as the one we inject, can be retraced one-to-one to the state of the device at the hardware level.

Lately, it has been demonstrated that transmon qubits are incredibly susceptible to natural radiation, that, by depositing charge in the substrate, breaks Cooper pairs and releases quasiparticles. These quasiparticles can tunnel the Josephson junction, exciting the qubit(s), and thus suddenly shifting their state(s) [14], as shown in Fig. 1(b)—upper). Numerous recent studies on the interaction of ionizing radiation with transmon-based devices have all highlighted a significant reduction in the correctness and interpretability of quantum circuit output [9], [10], [11], [12], [13], [14], [15], [28]. Recent experiments by Google Quantum AI, furthermore, demonstrate that the charge deposited by external radiation spreads across the quantum chip, suddenly modifying the state of multiple correlated physical qubits at once [8], as depicted in Fig. 1(b)—lower). In an even more recent study revolving around surface codes, the same Google Quantum AI team was forced to statistically outrule a *high-energy event* [29].

The rapid spread of charge across the qubits and the fault incidence rate have proven to be extremely significant, leading to a transient fault duration that has been measured to be of several milliseconds, which, as already mentioned, is orders of magnitude longer than a single-circuit multiple-shot execution. Google's experiment measured that a radiation-induced corruption event happens every tens of *seconds* on an array of just 25 physical qubits. To put this error rate in perspective, *hours* are needed to observe a radiation-induced event in a tens of thousands nodes traditional supercomputer [30].

Observation 1. The radiation-induced fault rate of qubits is orders of magnitude higher than the one of traditional transistors.

Unfortunately, the known QEC approaches, such as surface codes [27] or the Shor error correcting code [31], become ineffective when multiple physical qubits are affected by radiation [29]. If multiple logical qubits are mapped on the physical qubits of a single chip (as in most cases), we can expect one impinging particle to modify the state of multiple logical qubits. Stochastic and unpredictable radiation-induced faults; then, add over the intrinsic noise and suddenly modify the logical qubit state. As a result, even if QEC was

implemented, we would still expect logical-shift faults in quantum circuit executions making our evaluation valid also for future quantum machines.

These evidences highlight that stochastic particle strikes could possibly hinder large-scale use of QCs. More than ever it is now time for experts in the reliability domain to tackle the threat posed by such faults.

Observation 2. Logical-shift errors can occur in current NISQ machines, as radiation-induced faults suddenly change the qubit(s) quantum state and are not corrected by QEC approaches, since the deposited charge induces correlated faults in multiple physical qubits.

Regrettably, efficient and effective techniques to preserve the circuit output in case of logical-shift faults are still lacking. A possible approach to reduce the impact of radiation could be to shield quantum devices in deep underground caves [10], as recently announced by Oak Ridge National Lab and Fermi National Lab of the US Department of Energy [32]. Another option would be to replicate quantum chips, but the redundant chips, to maintain quantum properties, should share a quantum network and should be able to entangle qubits among different chips [33]. Both approaches are extremely resource intensive and expensive, and thus will hardly be the solution.

C. QUANTUM MACHINE LEARNING

Quantum machine learning (QML) explores how to devise and implement efficient quantum circuits that offer advantages over classical machine learning algorithms [34], [35]. The classical machine learning neuron operation is encoded in a binary fashion as *active* or *resting*, which could intuitively be translated to the basis states $|0\rangle$ and $|1\rangle$ of a qubit. This theoretically allows learning models to exploit quantum features, such as superposition and entanglement, possibly providing speedups or new processing approaches [6], [21], [36].

Li et al. [37] presented an exciting and long-awaited application of quantum multiplicative weight primal–dual ideas in supervised machine learning, achieving a quadratic improvement over classical counterparts. In addition, Kerenidis and Luongo [38] proposed quantum classification via slow feature analysis, while Havlíček et al. [39] developed and tested fully quantum neural networks, such as quantum support vector machines, on real quantum hardware, showing how an ever-increasing number of approaches are being adapted and tested with success in the QC field. Recently, also CNNs have been mapped on quantum circuits. The quantum convolutional layer (quanvolutional layer or qLayer for short) encodes a convolution kernel and a max pooling operation in the structure of a bounded-error quantum polynomial time circuit, called hardware efficient ansatz, and applies it to local subsections of an input, producing an output of higher level features. The substitution of a classical convolutional layer with a qLayer maintains the accuracy unaltered (since the two layers perform a comparable operation), but the network

with the qLayer still presents a lower loss and a faster convergence [6], [7], [21], [36], [40]. The models proposed in these papers still use the concept of the hardware efficient ansatz circuit, that we extensively analyze in this article, to derive the quantum layer. The detailed reliability evaluation of the quantum layer together with the fault effect characterization we propose can be directly applied to most of the available QNN models. To showcase how our results and observations can be used to evaluate the quantum fault propagation in QNN models, we target its original implementation [6] as a specific case study. We perform an exhaustive fine-grain fault injection campaign considering three incrementally complex versions of the original design, so as to let the reader compare the results with traditional convolution fault propagation. A generic hybrid architecture example is depicted in Fig. 2. The input image is divided into subgrids and both convolution and pooling on the image are performed through a four-qubit quantum circuit. The combination of all subgrids is the output feature map that is propagated to the downstream layer. As our results demonstrate, logical-shift faults as the one caused by intrinsic noise or natural radiation, can potentially corrupt the output prediction, therefore justifying the reason for studying faults' impact in QNNs.

III. EXPLORATION OF DESIGN SPACE

To have a thoughtful understanding of logic-shift error propagation, we propose a bottom–up approach, starting from a per-qubit reliability characterization of the qLayer circuit, to later consider the fault propagation in the QNN and its impact on the final classification. We study three network designs with incremental depths and two datasets. The hereby proposed methodology can be adapted and easily applied to test fault propagation in any other QML model, although such extensive analysis exceeds the scope of this article. We highlight several aspects that can impact the fault effect on the QNN operation, from the dependence of error propagation with the input image to the vulnerability of different qubits and different subgrids (position of the corrupted quanvolution in the feature map). We consider only faults affecting the quantum part of the QNN, thus no fault has been introduced in the classical layers.

A. QUANVOLUTIONAL LAYER

The first evaluation we propose is the characterization of the reliability profile of the ansatz four-qubit quantum circuit implementing the qLayer, shown in Fig. 3. The objective of this first exclusively quantum analysis is that of understanding the inner workings of fault propagation in this quantum circuit. The qLayer is composed of three main sections: encoding, the actual random circuit, and measurement. The sequence of these elements produces an output tensor of size comparable with a classical convolution *and* pooling operator on 2×2 subgrids with a stride of 2. The qLayer is not a direct quantum translation of the convolution operation for CNNs, but rather it is the standard quantum dual of a convolution kernel for QNNs, as per the works of the authors in [22], [23], [24],

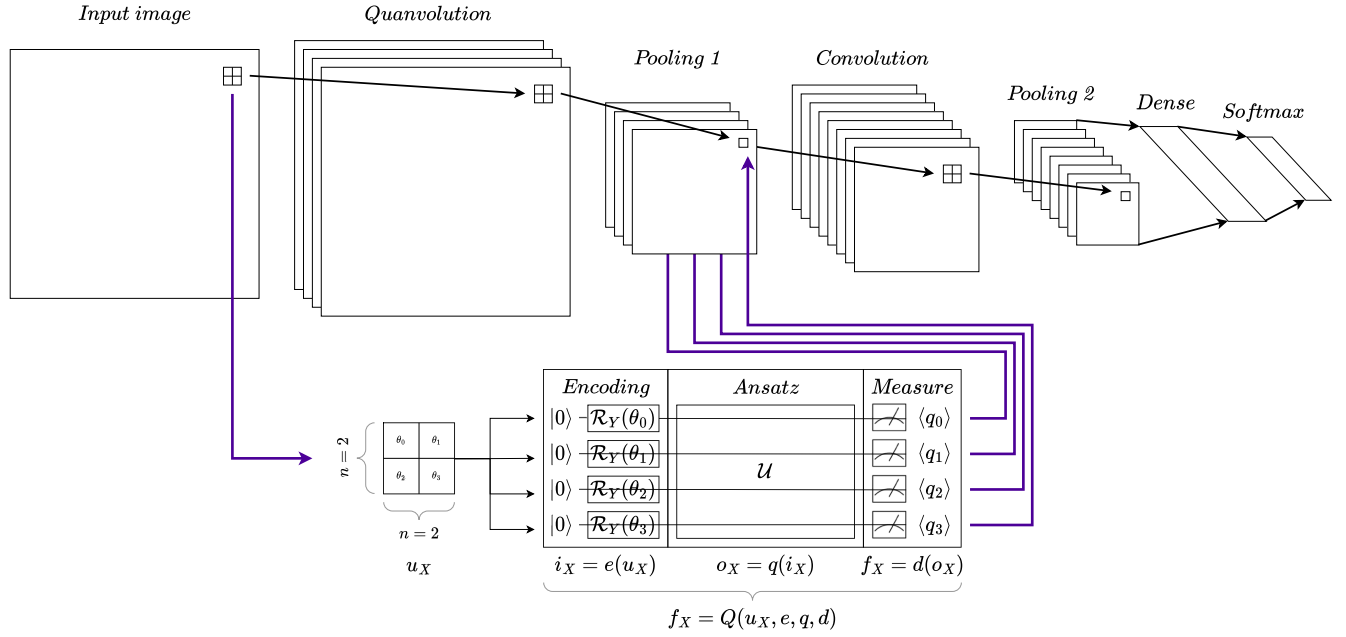


FIG. 2. Generic architecture of a hybrid QNN, with details of the qLayer. The input image is divided into 2×2 subgrids. A four-qubit quantum circuit performs both a 2×2 convolution and pooling operation on each subgrid. The output of the qLayer is a tensor of four channels representing the extracted feature map.

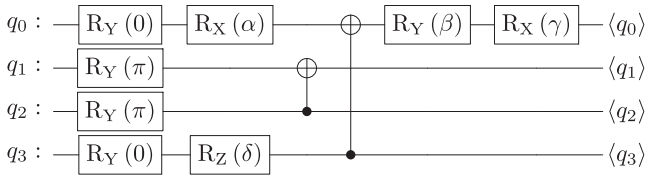


FIG. 3. Ansatz circuit implementing quanvolution. The values that parameterize the gates are randomly generated and kept constant during the experiments. The values fixed for the experiments are as follows: $\alpha = 2.353$, $\beta = 4.599$, $\gamma = 3.761$, and $\delta = 5.974$.

[25], [41], [42], and [43]. Each of the four qubits calculates one of the four channels of the feature map. Larger qLayers are possible, but in all the available QNN implementations the size of the subgrids is kept to 2×2 , which provides the best tradeoff between accuracy, circuit complexity, and performance [6], [21]. Thus, for this article, we keep constant the qLayer size at 2×2 . The proposed methodology and the following insights can be applied to any current and future qLayer sizes.

The circuit contains two controlled-NOT (CNOT) gates, each controlled by qubits 2 and 3, respectively, targeting qubits 1 and 0. The CNOT gate, a multiqubit gate, will perform an X-gate (the equivalent of the NOT gate in classical computing) on the target qubit if the state of the control qubit is $|1\rangle$. Given the entanglement caused by these two gates, we test also the propagation of faults from control qubits to target ones. At the end of the circuit, the expectation of each qubit is extracted, by running the circuit on a minimum of 1024 shots.

To have a fine-grain evaluation of the reliability of the quanvolution operation, in the analysis carried out in Section V-A, we have considered a fixed input subgrid and injected a fault in each of the four qubits (one qubit corrupted

at a time). The aim of the per-qubit evaluation is to understand which channel (qubit) is less reliable and if there is a difference between control and target qubits. As detailed in Section V-A, we found that faults in control qubits have a more significant impact on the QNN output, since they get propagated to the target qubit, and that the injection on one qubit affects only the channel associated with the corrupted qubit, with negligible effects on the other channels.

B. QNN AND INPUT DATASET

To understand how faults occurring in the qLayer propagate in the QNN, we make use of the knowledge gained from the previous in-depth analysis results on the qLayer, testing three hybrid models, so as to let the reader make a direct comparison with the well-studied fault propagation mechanisms of convolution layers in classical CNNs. We inject logic shifts only at inference time, not during training, which is a common practice also in traditional CNN reliability evaluation [18], [19]. In fact, while errors during training can potentially reduce the performance or increase the convergence time, these effects are easily detectable and solved with additional training steps. On the contrary, silent errors during inference can lead to potentially harmful real-time mispredictions and should be strictly avoided.

Recent experiments showed that the charge deposited by radiation migrates in the Silicon substrate, eventually affecting physically close qubits [8], [44]. Since the four qubits implementing the qLayer must be connected and close to each other, we expect the single-particle interaction to corrupt all of them. As such, in the QNN reliability evaluation, we will simultaneously corrupt all four qubits during subgrid computation.

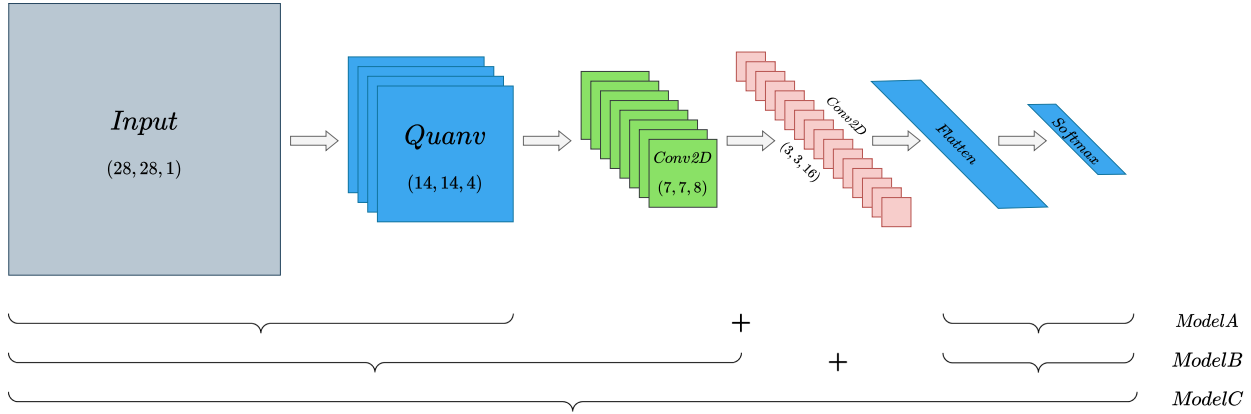


FIG. 4. Combined view of the three hybrid quantum neural networks studied. *ModelA* is composed of only a qLayer, directly connected to the Flatten layer. *ModelB* integrates a traditional convolution layer between the qLayer and the Flatten layer. *ModelC* is composed of all the elements in the figure above, i.e., a qLayer followed by two convolutional layers. Proportions for the output tensor dimensions have been preserved.

As a baseline, we consider the QNN design available in [6], a hybrid classical–quantum adaptation of the Le-Net model [45] for image classification, which is one of the first (classical and quantum) models to be designed. The inputs we used are taken from the MNIST handwritten digits and fashion datasets [46], both consisting of 70 000 28×28 pixels grayscale images representing either handwritten digits or clothing apparel.

We chose the MNIST datasets (handwritten digits and fashion items) since they are widely regarded as a cornerstone of classical machine learning (ML) research. In addition, the current scale of quantum devices does not yet allow for the usage of state-of-the-art, high-resolution datasets. Nevertheless, the results and insights provided by our analysis are still fundamental for characterizing the analyzed quantum design.

The QNN receives, as input, grayscale images with values ranging between 0 and 255. For each 2×2 subgrid in the input image, each pixel is encoded using amplitude embedding through a parameterized rotation \mathcal{R}_Y around the Y -axis, mapping each value linearly to the range $[0, \pi]$. In the qLayer, the quanvolution circuit is executed for each subgrid and the resulting tensor is propagated to the downstream layers.

We trace the fault propagation during the QNN inference and measure its impact on the output correctness. We distinguish between masked faults (the output is unaffected), tolerable silent data corruptions (the output is altered, but the correct class is selected), and misclassifications.

To have an overview of possible logical-shift errors propagation, we perform an exhaustive fault injection (more than 273 646 592 faults per image) in at least 30 random images from each dataset. In other words, once we have selected the injection site (qubit, channel, grid, etc.), we perform a complete fault injection, considering all the possible parameterized rotations, for each input image. Then, to understand the impact of error propagation from the input frame, we perform further experiments on 100 images. We have not

observed a significant dependence of fault propagation with the input image class.

C. QNN MODELS

The error propagation in classical CNNs is known to be dependent on the network depth (i.e., the number of layers the fault needs to traverse to reach the output) [18], [19]. In particular, convolution tends to spread the faults happening in upstream (traditional) layers. With the aim of understanding the dependence of logical-shift error propagation on the network depth, we consider three designs of increased complexity of the same QNN (based on [6]), hereby called *ModelA*, *ModelB*, and *ModelC*.

ModelA, whose structure is represented in Fig. 4, is the quintessential QNN, composed of the minimum number of layers. The qLayer takes as input a $(28, 28, 1)$ tensor and outputs a $(14, 14, 4)$ tensor. The latter is flattened and redirected into a Softmax dense layer.

From the barebone *ModelA*, we derive *ModelB* and *ModelC*, which are obtained by adding, respectively, one and two cascaded Conv2D operators between the qLayers and Flatten layers. The concatenation of a qLayer with classical convolutional layers has been done following state-of-the-art approaches in literature [6], choosing suitable filter sizes for the classical layers in the networks: each additional Conv2D layer doubles the number of filters used in the preceding operator and uses a filter size of 2×2 , with a stride of 2. It is worth noting that we do not consider multiple cascaded qLayer applications, following the approach in [6].

Each of the derived designs has been retrained to adapt the weights to the network depth. The accuracy on both the training and validation datasets obtained after the training of the three QNN designs is similar (at most 3% of difference) and comparable with the performance of the corresponding fully classical implementation.

Interestingly, as we detail in Section V-D, increasing the depth of the QNN by adding cascaded traditional convolutional layers *reduces* the quantum transient fault impact on

the output, masking some faults and reducing the probability to have misclassifications.

D. SINGLE AND MULTIPLE SUBGRID INJECTIONS

Finally, we also compare the reliability of QNNs when multiple subgrids are corrupted. In fact, the near-future prospect of highly integrated quantum chips prevented us from considering unrealistic the possible corruption of multiple subgrids at inference time, as detailed in Section V-E. For this reason, we also conducted experiments injecting on two distinct subgrids. As we show, when multiple subgrids are corrupted, the impact on the QNN's output is higher, increasing the probability of having misclassifications.

IV. EXPERIMENTAL SETUP

In this section, we describe the setup of the conducted experiments, providing details on the framework used to model the transient fault's effect.

A. LOGICAL-SHIFT ERROR MODEL

Fault injection in quantum circuits is more complex than in classical CMOS devices. In fact, the classical bit has only two states (0 and 1) and, thus, a bit-flip fault model is sufficient to study the reliability of CMOS devices. As seen in Observation 2, for qubits in a superposition, the interaction of ionizing particles can modify the quantum state by inducing a parameterized rotation (changing the ϕ and/or θ angle in the Bloch sphere, refer to Fig. 1). The magnitude of such parameterized rotations depends on the deposited charge, as shown with simulations [15] and experimentally validated [10], which can range from *meV* to *GeV* [47]. Thus, in contrast to classical computing, the quantum fault model has to take into account many more possible state changes than a "simple" bit flip (i.e., the X-Pauli gate), as a particle impact can induce *any* given parameterized rotation.

Since the energy of the impinging particle is continuous in a wide range (meV to GeV) [48], the fault's rotation range will also be continuous. As such, we consider all parameterized rotation magnitudes in our fault injection. This makes for a systematic analysis, which is as general as possible, without being tied to a specific particle energy range. The fault model and the results hereby presented can be easily weighted or normalized once more information on the correlation between exact impinging particle energy and fault amplitude will be known.

B. LOGICAL-SHIFT INJECTION AND SIMULATION

In this article, we only consider faults affecting the quantum part of the QNN. The effect of faults in the classical parts of CNNs has already been investigated deeply [18], [19], [20]. The simulations have been carried out without considering a device-level noise profile, as it is a well-separated event with respect to particle impacts, and its effects would add up to those of transient faults. In addition to this, we recall that noise has close to no impact on the ansatz circuit, as previously stated in Section III.

To inject logical-shift errors into the quantum convolution circuit during the QNN inference (we do not inject during training), we apply a tuned stimulus to modify the qubit state. To model the injected fault that, as discussed in Section II-B, can have parameterized rotations of different magnitudes, we use the QuFI fault injector, which inserts an extra $U3$ gate to model the fault [16]. The $U3$ gate can modify the ϕ and/or θ angles used to define the qubit's actual state (refer to Fig. 1). The ϕ angle modifies the phase of a qubit, and the θ angle changes the $|0\rangle - |1\rangle$ probability. The possible range for each angle without state duplication are $\phi = [0, 2\pi]$, and $\theta = [0, \pi]$. We also make a discretization of the angles range using a $\pi/12$ step size, which results in 325 possible configurations (i.e., distinct fault magnitudes to be injected).

To track fault propagation in QNNs, we broaden the applicability spectrum of the open source QuFI by porting it to the PennyLane [49] framework. We also achieve the possibility of running quantum circuits on devices provided by different vendors implementing different technologies, not to be limited to IBM machines, and a more direct QML-oriented development, since PennyLane inherently supports multiple libraries dedicated to the task.

The updated version of QuFI is part of our contribution and will be released as open source to stimulate further research in QNNs reliability.

C. FAULT EFFECT EVALUATION

As previously stated, the quantum circuit output is probabilistic, with each possible state having a certain probability to be selected. For instance, a two-qubit circuit has four possible states: $|00\rangle$, $|01\rangle$, $|10\rangle$, and $|11\rangle$. Ideally, the correct state will have the highest probability so it can be selected as the output. We use the quantum vulnerability factor (QVF) [50] metric to measure the impact of a transient fault in the output probability distribution. The QVF, corresponding to the architecture vulnerability factor [51] and the program vulnerability factor [52] in traditional computing systems, ranges from $[0,1]$, and indicates the probability of a fault to propagate affecting the output. In other words, the QVF indicates how likely the fault is, given the probabilistic output, to induce the selection of a corrupt state. A QVF close to zero indicates a high probability of selecting the correct state. Values close to one indicate that an incorrect state is likely to be selected. QVF values around 0.5 mean that the correct state and at least one incorrect state have similar probabilities, which makes the identification of correct states dubious.

To evaluate the effect of the propagation of logical-shifts in the qLayer to the downstream layers we measure also the *misclassification* rate of the tested QNNs. We inject faults into the qLayer during inference and let the corrupted output feed the downstream operations. Then, we check if the classification of the faulty execution is different than the classification of the fault-free one. We do not compare the faulty classification with the ground truth since we want to measure the impact of faults in the execution of a QNN. The (unlikely)

event of a fault improving accuracy is purely stochastic and not scientifically relevant, as we cannot rely on radiation to improve the QNN's accuracy. Moreover, we never observed such an event.

V. CHARACTERIZATION RESULTS

In this section, we detail the experimental results obtained from 13 322 547 200 logical-shift fault injection simulations (267 233 quantum circuit injections per input image, per configuration). This extensive evaluation provides a very accurate evaluation, with the statistical error being lower than 1% [53]. Our bottom-up evaluation starts from the characterization of the reliability of the quanvolution circuit, then understanding the fault effect on the QNN's output, and identifying how many faults induce misclassification. Then, we consider the QNN's reliability dependence on the dataset, the input image, and the subgrid. Finally, we evaluate how faults propagate in three different QNN designs of increasing complexity (ModelA, ModelB, ModelC) and we also discuss the impact of double faults. Our complete set of results is available in a public repository [54].

A. QLAYER RELIABILITY

As a first reliability evaluation, we detail the propagation of logical-shift faults in the quantum computation core of QNNs, that is, the qLayer implemented with the ansatz circuit depicted in Fig. 3. For this evaluation, we consider the qLayer as a standalone quantum circuit, i.e., without the integration with the upstream and downstream portions of the QNN. To have a fine grain understanding, we inject in each qubit separately.

To assess the resilience profile of the circuit, we use as input a fixed 2×2 subgrid, with the top-right-hand side and bottom-left-hand side pixels as white (value 255) and the other two as black (value 0), i.e., a diagonal black and white subgrid. This corresponds to encoding qubits 0 and 3 of Fig. 3 in state $|0\rangle$, while qubits 1 and 2 are encoded in state $|1\rangle$, since they are prepared by rotations around the Y-axis of 0 and π radians, respectively.

In Fig. 5 we plot, for each (θ, ϕ) logical-shift, the QVF for the qLayer circuit, increasing the logical-shift in θ (0 to π) and ϕ (0 to 2π). We inject in each qubit separately. A QVF close to 1 (red) indicates a shift that entails an high probability of selecting the wrong output, while values close to 0 (green) indicate shifts that do not modify the output selection.

In Fig. 5, we can see that the QVF increases (worsens) as we move to the right-hand side of the picture, while it is almost unaltered as we move up in the picture. This means that the qLayer circuit becomes highly affected by the azimuthal faults (θ logical-shift) for values greater than $\pi/2$. While this result might seem obvious and intuitive (a higher modification leads to a higher impact on the output), it has been shown that for quantum circuits logical-shifts of higher magnitude do not necessarily have a higher probability to modify the circuit output [16].

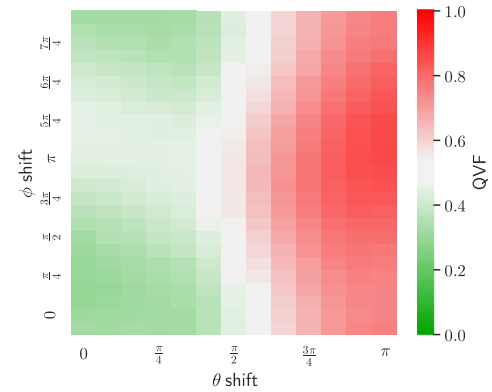


FIG. 5. QVF (probability for a fault to modify the output correctness) heatmap for *single* logical-shift fault injections in the circuit implementing the qLayer. We inject θ logical-shifts from 0 to π and ϕ logical-shifts from 0 to 2π in one qubit.

Interestingly, the qLayer shows a relatively low vulnerability to the polar angle (ϕ), albeit a small QVF rise between $3\pi/4$ and $5\pi/4$. Analyzing the details of the single-qubit QVF heatmaps (not reported here but included in the public repository [54]), we found that qubits 0 (target) and 3 (control) are responsible for lowering the average resilience of the circuit for $0 < \theta < \pi/4$ and $3\pi/4 < \phi < 5\pi/4$ (white region). This is because these two qubits undergo more quantum gates than qubits 1 and 2.

The QVF heatmap suggests that the θ shifts are critical, whereas ϕ logical-shifts are not. We will further investigate this property at the network level in the Section V-B.

Observation 3. Due to the usage of amplitude embedding, ϕ logical-shifts do not significantly modify the qLayer output, whereas θ shifts cause an effect on the output that is proportional to the shift magnitude.

We have also observed that a single injected fault in a qubit of the qLayer circuit modifies all its logically connected qubits, and consequently the output bit string. This means that the computation of the qLayer is likely to spread the fault, corrupting the cascaded layers in the network's architecture.

Observation 4. A fault in a single qubit of the qLayer spreads to all its logically connected qubits.

B. FAULT PROPAGATION IN QNNs

To understand how faults propagate in QNNs and identify the faults that generate misclassifications, we perform an extensive fault injection campaign injecting a logical-shift fault in each of the four qubits executing one quanvolution (i.e., calculating one subgrid). We consider all three network models, with an increasing depth, on both input datasets and over single and double subgrids injected. Faults that did not corrupt the Softmax vector output of the neural network have been labeled as *masked*. Faults that modified the output vector have been labeled as either *tolerable* if they did not alter the output predicted class, or *misclassified* otherwise.

We have observed that *all* of the θ logical-shifts propagate to ModelA's output (not necessarily modifying the

TABLE 1 Phase-Shift Fault-Induced Misclassification Probability

QNN design	data set	Single grid	Double grid
ModelA	digits	3.49%	6.05%
	fashion	1.23%	1.94%
ModelB	digits	5.52%	10.65%
	fashion	3.99%	6.58%
ModelC	digits	1.69%	8.62%
	fashion	3.16%	6.33%

classification) whereas *none* of the injections of ϕ logical-shift causes an observable effect on the network output. The fact that the injections of ϕ logical-shift do not propagate should not surprise. As discussed in Section IV, the qLayer circuit uses amplitude embedding, i.e., maps the convolution data in the θ angle of the qubit state, the $|0\rangle - |1\rangle$ probability. Thus, changes to the phase (ϕ angle) of a qubit state are expected to have a small impact on the qLayer output (as confirmed in Observation 4) and, as our fault injection in the QNN shows, ϕ polar shifts do not modify the inference. In the following, we only report θ shift injections.

Observation 5. In a simple QNN with just one qLayer, no ϕ logical-shift modifies the output but all θ logical-shifts propagate to the output.

Table I gives the measured average probability among all the logical-shift faults injected in the qLayer circuit to induce a misclassification across all the possible configurations of datasets, models, and the number of subgrids injected at a time. Our analysis shows that the misclassification rate can vary from 1.23% to up to 10.65%, depending on the QNN design and dataset. This misclassification probability is the result of the interaction of a plethora of factors: in the next sections, we go into the details of the dependencies of the misclassification rate from the logical-shift magnitude, network design, and the number of simultaneously injected subgrids.

The measured misclassification rates for QNNs, given in Table 1, are comparable with the ones of classical CNNs, that range from 1% (floating point) to 7% (with a specific fixed-point data type) [18]. From Observation 1, we know that the CMOS error rate is *orders of magnitude* lower than the one of a superconducting transmon qubit. Thus, while CNNs and QNNs have similar misclassification probability, the latter are much more likely to experience a fault (see Section II-B) and will experience a considerably higher misclassification rate.

Observation 6. The probability for a fault to generate a misclassification in a QNN or in a CNN is comparable. However, in QNN, the fault rate is orders of magnitude higher.

In Figs. 6 and 7, we provide, respectively, an example of the effects of a *tolerable* fault and of a *misclassification* fault on the Softmax vector output. To better understand the effect of fault propagation, in Fig. 6, we show an example of a fault that does not induce misclassification while modifying significantly the classes' probability distribution. The

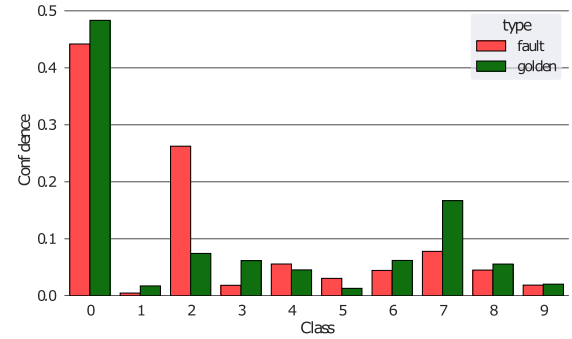


FIG. 6. Example of a fault that does not induce misclassification. We plot the Softmax layer outputs for the baseline fault-free (in the figure, labeled as golden) execution (in green) and faulty execution (in red) obtained by injecting a fault amplitude of $\theta = \frac{\pi}{2}$ in all qubits of the qLayer processing a subgrid. Despite class 2 confidence increasing significantly, class 0 is still correctly classified.

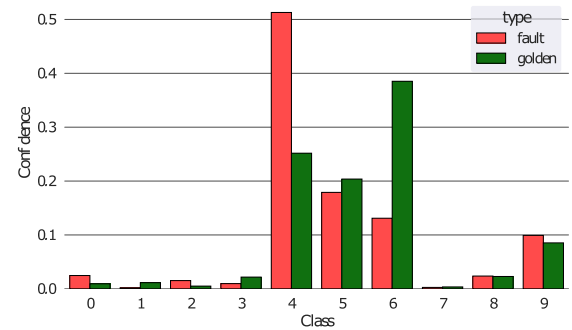


FIG. 7. Example of a fault that induces misclassification. We plot the Softmax layer outputs for the baseline fault-free (in the figure, labeled as golden) execution (in green) and faulty execution (in red) obtained by injecting a fault amplitude of $\theta = \frac{\pi}{2}$ in all qubits of the qLayer processing a subgrid. The fault increases the confidence of class 4, promoting it to the selected output class, while class 6 (the correct classification) lags behind as the third most probable output.

plotted data refer to a $\theta = \frac{\pi}{2}$ fault injected in all four qubits of the qLayer applied to a single subgrid out of the 196 possible subgrids of the input image. In the baseline fault-free execution, class 0 is selected with very high confidence (0.48 versus 0.18 of the second class). The fault triplicates the confidence for class 2 to be selected while reducing the one for class 0. Nonetheless, despite a significant reduction in the classification confidence (0.44 of class 0 versus 0.26 of class 2), class 0 is still the one with the highest probability.

In Fig. 7, we show an example of a misclassification fault. The baseline fault-free execution classifies the input as class 6, but with a low confidence (0.38), since both class 4 and class 5 have a high probability at the QNN output. The $\theta = \frac{\pi}{2}$ fault we inject in the qLayer reduces to 1/3 the probability of class 6 and doubles class 4 probability, eventually leading to misclassification.

C. MISCLASSIFICATION DEPENDENCE ON SUBGRID AND INPUT

To understand possible QNN reliability dependencies from the input frame and the corrupted subgrid, we have performed an extensive fault injection on 100 images for each

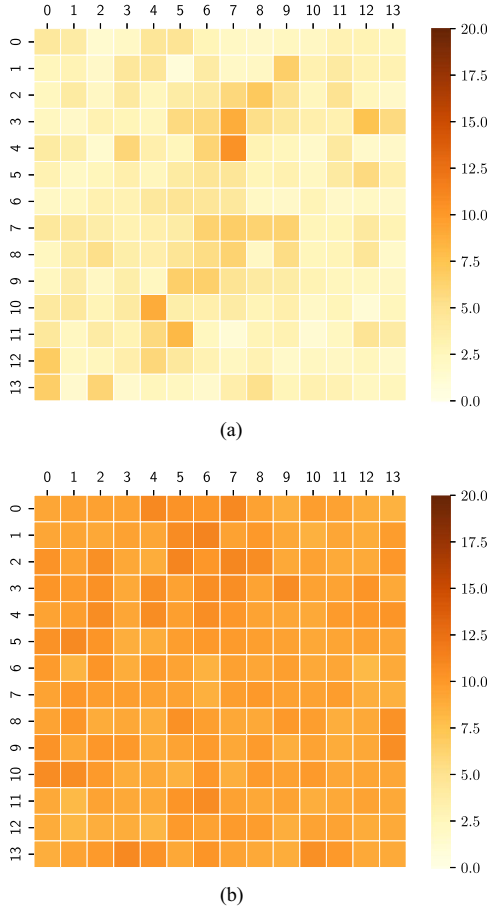


FIG. 8. Heatmaps showing the misclassification probability for faults injecting in each subgrid (identified by the coordinates in the images). Data have been obtained testing 100 images of (a) digits—handwritten digits dataset and (b) fashion datasets.

dataset and injected a fault in every single subgrid of the input image. Since each image has 196 subgrids, this campaign is computationally demanding to execute, requiring a total of more than seven billion injections for both datasets.

Fig. 8 shows the average misclassification probability for each subgrid on the (a) digits and (b) fashion datasets. To ease visualization, we plot the misclassification rate as a heatmap, where the (row, column) are the coordinates of the subgrid location. As can be seen by comparing Fig. 8(a) and (b), the two datasets have a completely different reliability dependence on the corrupted subgrids.

In the handwritten digits dataset, as shown in Fig. 8(a), some subgrids are extremely likely to generate misclassification while others, even if corrupted, have a low probability to impact the network output. For instance, the subgrid in (row: 4, column: 7) has a misclassification ratio of 10.3% whereas a fault in the subgrid (row: 1, column: 5) has a 0.8% probability to induce a misclassification. In the fashion dataset Fig. 8(b) the heatmap has a homogeneous distribution of misclassification ratios, suggesting that the probability of incorrectly labeling an image on this second dataset is not significantly dependent on the corrupted subgrid. Finally, we

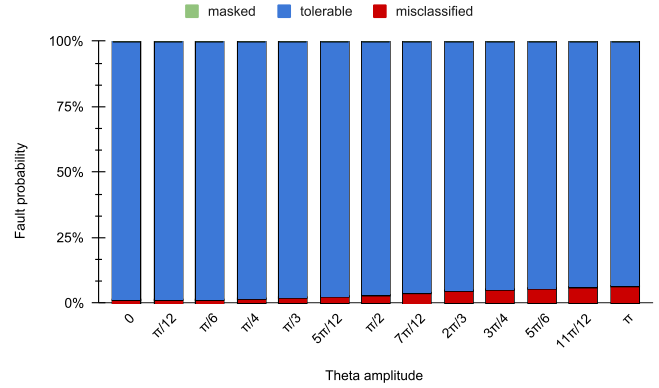


FIG. 9. Misclassification ratio with respect to fault angle amplitude θ on the barebone ModelA, considering a single failed subgrid, computed on the digits dataset.

have not registered an input image class dependence on the misclassification rate.

Observation 7. The misclassification probability depends on the corrupted subgrid in the digits dataset, while there is no dependence between misclassification and object class.

D. FAULT PROPAGATION DEPENDENCE ON QNN DESIGN

To understand if the QNN design impacts the fault propagation, we inject in a single *random* subgrid of the qLayer on ModelA (one qLayer), ModelB (one qLayer and one Conv2D layer), and ModelC (one qLayer and two Conv2D layers) with dataset partitions of size 30, to test how much downstream classical layer(s) impact the quantum fault propagation. Details about the three QNN designs can be found in Section III-C.

At first, we present the analysis of ModelA on the MNIST handwritten dataset partition, and plot in Fig. 9 the percentage of *misclassified*, *tolerable*, and *masked* faults with respect to the amplitude of the angle θ in the parameterizable U3 fault gate. There is an evident correlation between the amplitude of θ and the incidence of misclassifications in the network's output. Faults with an amplitude of just $\theta = \frac{\pi}{2}$ produce a 3.18% misclassification ratio, which bumps up to 6.43% for a fault amplitude of $\theta = \pi$. Moreover, given the relatively shallow architecture of ModelA, the classical part of the network cannot sufficiently compensate for the fault and no *masked* event is ever registered. All of the injected faults in fact produce a variation in the output Softmax vector.

In Fig. 10, once again computed on the handwritten digits dataset, ModelB undergoes a fault at the qLayer level, which gets propagated first through the Conv2D layer and later in the Flatten and Softmax layers. On a fault gate amplitude of $\theta = \frac{\pi}{2}$, the misclassification ratio is valued at 5.84%, rising to 7.39% when considering the maximum fault amplitude. Much like for ModelA, it is once again clear to see that there is a correlation between the azimuthal angle of the U fault gate θ and a rise in the misclassification ratio. No *masked* event has been observed. On average, as seen in Table I, the

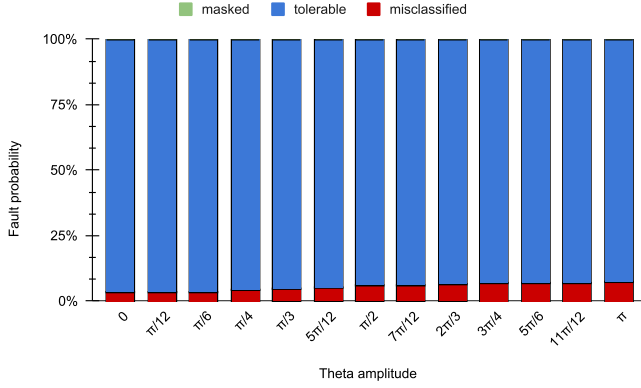


FIG. 10. Misclassification ratio with respect to fault angle amplitude θ on ModelB, considering a single failed subgrid, computed on the digits dataset.

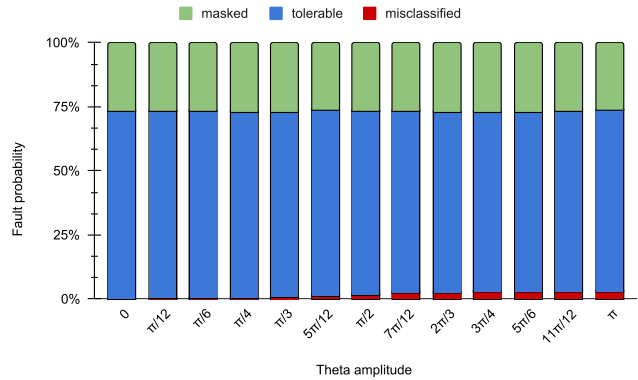


FIG. 11. Misclassification ratio with respect to fault angle amplitude θ on ModelC, considering a single failed subgrid, computed on the digits dataset.

misclassification ratio for ModelB is 5.52% on the handwritten digits dataset, while the same analysis on the fashion dataset boasts a slightly lower average rate of 3.99%. The average probability for ModelB to produce a wrong output class prediction increases, *w.r.t.* ModelA, by a significant margin in both datasets.

ModelC's reliability behavior is detailed in Fig. 11, once again on the handwritten digits dataset. Unlike the other experiments, we observe, on average, a stable distribution of *masked* events with a probability of 26.67%: this can be explained by the fact that the increasing number of filters in the Conv2D operators eventually disperses the effect of a portion of the faults introduced at the quantum layer and eventually those get canceled out by undergoing a product operation with weights or kernel parameters equal to zero.

Observation 8. Larger θ logical-shifts increase the misclassification probability, in all the tested QNN designs.

It is important to note that this event depends on the qLayer, as it is not the direct quantum translation of a convolution and thus boasts a different behavior. Moreover, a significant drop in the overall misclassification rate is observed, with average values of 1.69% for the handwritten digits dataset and a maximum registered at 3.17% at the highest fault gate amplitude of $\theta = \pi$. Similarly, on the fashion

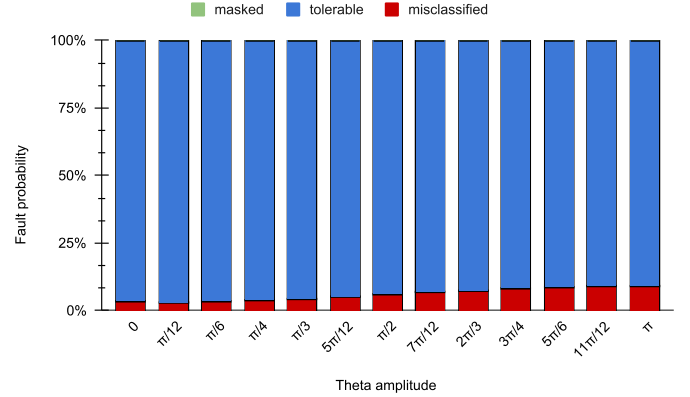


FIG. 12. Misclassification ratio with respect to fault angle amplitude θ on the barebone QNN model, considering two failed subgrids, computed on the digits dataset.

dataset, an average of 3.16% misclassifications is registered, with a *masked* events ratio of 26.59%.

Observation 9. Downstream Conv2D layers can help in masking some qLayer faults.

E. DOUBLE SUBGRIDS CORRUPTION

In a general quantum workload, we cannot rule out the possibility to experience multiple radiation-induced corruptions across the whole execution, especially in iterative approaches, such as QNNs or in deep quantum circuits. CMOS devices, in terrestrial applications, can be corrupted mostly by neutrons and the probability for a CMOS-based chip (even large GPUs) to be corrupted by an impinging neutron is very low, in the order of 10^{-6} – 10^{-8} [17], [55]. Since the flux of neutrons at sea level is about $13 \text{ n/cm}^2/\text{h}$, the error rate of a CMOS chip is in the order of 10^{-5} – 10^{-9} errors per hour [17], making it highly unlikely to observe two events in a single computation. Unfortunately, this does not hold for qubits, since they have an intrinsic coherent time in the order of ms and a sensitivity to radiation that is much higher than CMOS transistors (Observation 2) and, moreover, they can be affected by various uncorrelated radiation sources (neutrons, muons, etc.) [8], [10]. In addition, we expect quantum chips to be highly integrated in the near future, possibly including multiple qLayer circuits (Observation 1) on a smaller surface area. As a result, we can expect to have the single-particle deposited charge corrupting multiple logical qubits or possibly even multiple qLayer circuits.

Therefore, as a final analysis, we have injected in two separate random subgrids at the same time across all QNN models and input datasets. The results presented in Fig. 12 refer to ModelA on the handwritten digits dataset partition of size 30. Similarly to the case when a single subgrid is corrupted, a correlation between the amplitude and the misclassification ratio is evident, where a fault amplitude of $\theta = \frac{\pi}{2}$ is responsible for changing the output predicted class in 6.24% of cases, almost doubled with respect to the previous experiment on single-subgrid injections. The misclassification ratio tops out

at 9.0% with the highest amplitude injection of $\theta = \pi$. We did not observe any *masked* injections.

Additional experiments obtained by testing double subgrid injections on both ModelB and ModelC have been performed, boasting a steady increase in the rate of misclassification events. Moreover, ModelC undergoes a reduction in the number of *masked* events when the number of injected subgrids is doubled. The average rates for these experiments are reported in Table I.

Observation 10. The corruption of two subgrids significantly increase the misclassification probability.

Complete access to the data regarding all these experiments, which have not been further commented on here due to lack of space, is available in [54].

VI. DISCUSSION AND PROJECTIONS

The QNN architecture we have characterized is the first model of its kind ever proposed [6]. This design is the cornerstone over which rapidly growing and vibrant research is being carried out [22], [23], [24], [25], [41], [42], [43]. In particular, the structure of the hardware efficient ansatz we have deeply investigated is being used to implement *quanvolution* in the vast majority of QNNs models. For this reason, our tool and analysis results can be used to understand the reliability behavior of current and future QNNs making use of the same qLayer or other layers derived from it according to their shared characteristics.

Thanks to the continuous advancements in QC technology, the application landscape for QNNs keeps broadening. As we have shown; however, the widespread adoption of QC could be stifled by logical-shifts caused by either intrinsic noise or cosmic rays, particularly on superconducting transmon quantum devices [8], [9], [10], [11], [12], [13], [14], [15], [16]. Despite the fact that QNNs have a misclassification ratio comparable with that of CNNs, their reliability is much more significantly hindered with respect to their classical counterparts, given that the radiation-induced fault rate for quantum devices is orders of magnitude higher with respect to CMOS. The usage of surface codes along scalability and construction quality improvements may have a positive role in improving the reliability of many QML models, at which point the hereby presented systematic results may simply be reweighted according to the way in which they impact the output distribution. At the moment; however, there is no guarantee that surface codes will not fail in the event of a particle impact, and may as well worsen the results in this circumstance.

Hardware/software co-design has been demonstrated to be critical for quantum computers [26], [56], [57], [58], [59], [60], [61]. Our analysis adds the logical-shift fault issue to the reliability assessment of these devices and architectures. This work's results, alongside the methodology employed, can direct algorithm design, innovative software/hardware hardening solutions development, and more robust circuit architecture implementation. For instance, quantum circuit designers could leverage our framework to implement and

test purposefully made QEC codes, adding redundancy in the most critical part or duplicating only the most critical *quanvolutions*, and thus largely reducing the misclassification ratio. The information regarding subgrid criticality can help, knowing the dataset used in the field, in designing a future scheduler or optimizer for QML workloads to map each subgrid execution onto more or less reliable quantum hardware with respect to their impact in case of a fault. Moreover, we envision that transpilers may exploit our analysis through an additional heuristic metric, aimed at reducing the impact of radiation-induced faults, and adaptable to any physical quantum device. Finally, our analysis highlights that better training or a different QNN design might increase the classification confidence and reduce radiation-induced misclassifications, nonetheless this can hardly solve the faults issue altogether.

VII. CONCLUSION

In this article, we have proposed a methodology to deeply investigate the propagation of logical-shift faults in QNNs. By using a fault model derived from experiments and simulations, we demonstrate that the corruption of the qLayer significantly impacts QNNs' operations and classification. Our data show that θ logical-shifts are very likely to propagate in the QNN, and that up to 10% of injections induce misclassification. As we have seen, the misclassification probability depends on the logical-shift magnitude, on the corrupted subgrid, on the dataset, and on the number of classical layers that follow the corrupted layer.

In the future, we intend to propose mitigation or hardening solutions for QNNs. We aim at blocking the fault propagation in the qLayer and reducing its probability to cause a misclassification.

REFERENCES

- [1] D. Herman et al., "A survey of quantum computing for finance," 2022, doi: [10.48550/arXiv.2201.02773](https://doi.org/10.48550/arXiv.2201.02773).
- [2] A. Peruzzo et al., "A variational eigenvalue solver on a photonic quantum processor," *Nature Commun.*, vol. 5, no. 1, 2014, Art. no. 4213, doi: [10.1038/ncomms5213](https://doi.org/10.1038/ncomms5213).
- [3] R. Mullin, "Let's talk about quantum computing in drug discovery," *C&EN Glob. Enterprise*, vol. 98, no. 35, pp. 20–22, 2020, doi: [10.1021/cen-09835-feature2](https://doi.org/10.1021/cen-09835-feature2).
- [4] S. Lloyd, M. Mohseni, and P. Rebentrost, "Quantum algorithms for supervised and unsupervised machine learning," 2013, doi: [10.48550/arXiv.1307.0411](https://doi.org/10.48550/arXiv.1307.0411).
- [5] H.-Y. Huang et al., "Power of data in quantum machine learning," *Nature Commun.*, vol. 12, no. 1, 2021, Art. no. 2631, doi: [10.1038/s41467-021-22539-9](https://doi.org/10.1038/s41467-021-22539-9).
- [6] M. Henderson, S. Shakyia, S. Pradhan, and T. Cook, "Quanvolutional neural networks: Powering image recognition with quantum circuits," *Quantum Mach. Intell.*, vol. 2, no. 1, 2020, Art. no. 2, doi: [10.1007/s42484-020-00012-y](https://doi.org/10.1007/s42484-020-00012-y).
- [7] K. Sooksatra, P. Rivas, and J. Orduz, "Evaluating accuracy and adversarial robustness of quanvolutional neural networks," in *Proc. Int. Conf. Comput. Sci. Comput. Intell.*, 2021, pp. 152–157, doi: [10.1109/CSCI54926.2021.00097](https://doi.org/10.1109/CSCI54926.2021.00097).
- [8] M. McEwen et al., "Resolving catastrophic error bursts from cosmic rays in large arrays of superconducting qubits," *Nature Phys.*, vol. 18, no. 1, pp. 107–111, Jan. 2022, doi: [10.1038/s41567-021-01432-8](https://doi.org/10.1038/s41567-021-01432-8).

- [9] A. D. Corcoles et al., "Protecting superconducting qubits from radiation," *Appl. Phys. Lett.*, vol. 99, no. 18, 2011, Art. no. 181906, doi: [10.1063/1.3658630](#).
- [10] L. Cardani et al., "Reducing the impact of radioactivity on quantum circuits in a deep-underground facility," *Nature Commun.*, vol. 12, no. 1, 2021, Art. no. 2733, doi: [10.1038/s41467-021-23032-z](#).
- [11] J. M. Martinis, "Saving superconducting quantum processors from decay and correlated errors generated by gamma and cosmic rays," *npj Quantum Inf.*, vol. 7, no. 1, 2021, Art. no. 90, doi: [10.1038/s41534-021-00431-0](#).
- [12] Z. Chen et al., "Exponential suppression of bit or phase errors with cyclic error correction," *Nature*, vol. 595, no. 7867, pp. 383–387, 2021, doi: [10.1038/s41586-021-03588-y](#).
- [13] L. Grünhaupt et al., "Loss mechanisms and quasiparticle dynamics in superconducting microwave resonators made of thin-film granular aluminum," *Phys. Rev. Lett.*, vol. 121, Sep. 2018, Art. no. 117001, doi: [10.1103/PhysRevLett.121.117001](#).
- [14] A. P. Vepsäläinen et al., "Impact of ionizing radiation on superconducting qubit coherence," *Nature*, vol. 584, no. 7822, pp. 551–556, 2020, doi: [10.1038/s41586-020-2619-8](#).
- [15] C. D. Wilen et al., "Correlated charge noise and relaxation errors in superconducting qubits," *Nature*, vol. 594, no. 7863, pp. 369–373, 2021, doi: [10.1038/s41586-021-03557-5](#).
- [16] D. Oliveira et al., "QuFI: A quantum fault injector to measure the reliability of qubits and quantum circuits," in *Proc. IEEE/IFIP 52nd Annu. Int. Conf. Dependable Syst. Netw.*, 2022, pp. 137–149, doi: [10.1109/DSN53405.2022.00025](#).
- [17] R. Baumann, "Soft errors in advanced computer systems," *IEEE Des. Test Comput.*, vol. 22, no. 3, pp. 258–266, May/Jun. 2005, doi: [10.1109/MDT.2005.69](#).
- [18] G. Li et al., "Understanding error propagation in deep learning neural network (DNN) accelerators and applications," in *Proc. Int. Conf. High Perform. Comput. Netw. Storage Anal.*, 2017, pp. 1–8, doi: [10.1145/3126908.3126964](#).
- [19] F. F. d. Santos et al., "Analyzing and increasing the reliability of convolutional neural networks on GPUs," *IEEE Trans. Rel.*, vol. 68, no. 2, pp. 663–677, Jun. 2019, doi: [10.1109/TR.2018.2878387](#).
- [20] Y. Ibrahim et al., "Soft error resilience of deep residual networks for object recognition," *IEEE Access*, vol. 8, pp. 19490–19503, 2020, doi: [10.1109/access.2020.2968129](#).
- [21] M. Schuld, I. Sinayskiy, and F. Petruccione, "The quest for a quantum neural network," *Quantum Inf. Process.*, vol. 13, no. 11, pp. 2567–2586, Aug. 2014, doi: [10.1007/s1128-014-0809-8](#).
- [22] T. Sathya and S. Sudha, "OQCNN: Optimal quantum convolutional neural network for classification of facial expression," *Neural Comput. Appl.*, vol. 35, pp. 9017–9033, 2023, doi: [10.1007/s00521-022-08161-w](#).
- [23] S.-Y. Huang, W.-J. An, D.-S. Zhang, and N.-R. Zhou, "Image classification and adversarial robustness analysis based on hybrid quantum-classical convolutional neural network," *Opt. Commun.*, vol. 533, 2023, Art. no. 129287, doi: [10.1016/j.optcom.2023.129287](#).
- [24] Y. Dong, Y. Fu, H. Liu, X. Che, L. Sun, and Y. Luo, "An improved hybrid quantum-classical convolutional neural network for multi-class brain tumor MRI classification," *J. Appl. Phys.*, vol. 133, no. 6, 2023, Art. no. 064401, doi: [10.1063/5.0138021](#).
- [25] R. Choudhuri and A. Halder, "Brain MRI tumour classification using quantum classical convolutional neural net architecture," *Neural Comput. Appl.*, vol. 35, no. 6, pp. 4467–4478, Oct. 2022, doi: [10.1007/s00521-022-07939-2](#).
- [26] G. Li, A. Wu, Y. Shi, A. Javadi-Abhari, Y. Ding, and Y. Xie, "On the co-design of quantum software and hardware," in *Proc. 8th Annu. ACM Int. Conf. Nanoscale Comput. Commun.*, 2021, pp. 1–7, doi: [10.1145/3477206.3477464](#).
- [27] S. Bravyi and A. Y. Kitaev, "Quantum codes on a lattice with boundary," *Quantum Phys.*, 1998. [Online]. Available: <https://api.semanticscholar.org/CorpusID:10722101>
- [28] R. Barends et al., "Minimizing quasiparticle generation from stray infrared light in superconducting quantum circuits," *Appl. Phys. Lett.*, vol. 99, no. 11, 2011, Art. no. 113507, doi: [10.1063/1.3638063](#).
- [29] R. Acharya et al., "Suppressing quantum errors by scaling a surface code logical qubit," *Nature*, vol. 614, pp. 7949, pp. 676–681, Feb. 2023, doi: [10.1038/s41586-022-05434-1](#).
- [30] D. Tiwari et al., "Understanding GPU errors on large-scale HPC systems and the implications for system design and operation," in *Proc. IEEE 21st Int. Symp. High Perform. Comput. Archit.*, 2015, pp. 331–342, doi: [10.1109/HPCA.2015.7056044](#).
- [31] P. W. Shor, "Scheme for reducing decoherence in quantum computer memory," *Phys. Rev. A*, vol. 52, pp. R2493–R2496, Oct. 1995, doi: [10.1103/PhysRevA.52.R2493](#).
- [32] L. Hesla, "Fermilab is partner in quantum science center based at Oak Ridge National Laboratory," Nov. 2020, doi: [10.1103/PhysRevA.52.R2493](#).
- [33] Q. Xu et al., "Distributed quantum error correction for chip-level catastrophic errors," *Phys. Rev. Lett.*, vol. 129, 2022, Art. no. 240502, *arXiv:2203.16488*, doi: [10.1103/PhysRevLett.129.240502](#).
- [34] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, no. 7671, pp. 195–202, Sep. 2017, doi: [10.1038/2Nature23474](#).
- [35] N. Killoran, T. R. Bromley, J. M. Arrazola, M. Schuld, N. Quesada, and S. Lloyd, "Continuous-variable quantum neural networks," *Phys. Rev. Res.*, vol. 1, Oct. 2019, Art. no. 033063, doi: [10.1103/PhysRevResearch.1.033063](#).
- [36] M. Monnet et al., "Pooling techniques in hybrid quantum-classical convolutional neural networks," in *Proc. IEEE Int. Conf. Quantum Comput. Eng.*, 2023, pp. 601–610, doi: [10.1109/QCE57702.2023.00074](#).
- [37] T. Li, S. Chakrabarti, and X. Wu, "Sublinear quantum algorithms for training linear and kernel-based classifiers," in *Proc. 36th Int. Conf. Mach. Learn.*, 2019, pp. 3815–3824, doi: [10.48550/arXiv.1904.02276](#).
- [38] I. Kerenidis and A. Luongo, "Classification of the MNIST data set with quantum slow feature analysis," *Phys. Rev. A*, vol. 101, Jun. 2020, Art. no. 062327, doi: [10.1103/PhysRevA.101.062327](#).
- [39] V. Havlíček et al., "Supervised learning with quantum-enhanced feature spaces," *Nature*, vol. 567, no. 7747, pp. 209–212, Mar. 2019, doi: [10.1038/s41586-019-0980-2](#).
- [40] A. Abbas, D. Sutter, C. Zoufal, A. Lucchi, A. Figalli, and S. Woerner, "The power of quantum neural networks," *Nature Comput. Sci.*, vol. 1, no. 6, pp. 403–409, Jun. 2021, doi: [10.1038/s43588-021-00084-1](#).
- [41] U. Ullah, A. G. O. Jurado, I. D. Gonzalez, and B. Garcia-Zapirain, "A fully connected quantum convolutional neural network for classifying ischemic cardiopathy," *IEEE Access*, vol. 10, pp. 134592–134605, 2022, doi: [10.1109/access.2022.3232307](#).
- [42] A. Senokosov, A. Sedykh, A. Sagingalieva, B. Kyriacou, and A. Melnikov, "Quantum machine learning for image classification," *Mach. Learn.: Sci. Technol.*, vol. 5, no. 1, Mar. 2024, Art. no. 015040.
- [43] A. Sagingalieva et al., "Hybrid quantum ResNet for car classification and its hyperparameter optimization," *Quantum Mach. Intell.*, vol. 5, no. 2, Sep. 2023, Art. no. 38, doi: [10.1007/s42484-023-00123-2](#).
- [44] D. Oliveira, E. Auden, and P. Rech, "Atmospheric neutron-induced fault generation and propagation in quantum bits and quantum circuits," *IEEE Trans. Nucl. Sci.*, vol. 70, no. 4, pp. 345–353, Apr. 2023, doi: [10.1109/TNS.2022.3226141](#).
- [45] Y. LeCun et al., "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989, doi: [10.1162/neco.1989.1.4.541](#).
- [46] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998, doi: [10.1109/5.726791](#).
- [47] L. Cardani et al., "Disentangling the sources of ionizing radiation in superconducting qubits," *Eur. Phys. J. C*, vol. 83, no. 1, Jan. 2023, Art. no. 94, doi: [10.1140/epjc/s10052-023-11199-2](#).
- [48] T. Roth, R. Ma, and W. C. Chew, "The transmon qubit for electromagnetics engineers: An introduction," *IEEE Antennas Propag. Mag.*, vol. 65, no. 2, pp. 8–20, Apr. 2023, doi: [10.1109/MAP.2022.3176593](#).
- [49] V. Bergholm et al., "PennyLane: Automatic differentiation of hybrid quantum-classical computations," 2018, doi: [10.48550/arXiv.1811.04968](#).
- [50] D. Oliveiray, E. Giusto, B. Baheriz, Q. Guanz, B. Montruccio, and P. Rechx, "A systematic methodology to compute the quantum vulnerability factors for quantum circuits," *IEEE Trans. Dependable Secure Comput.*, early access, Sep. 12, 2023, doi: [10.1109/TDSC.2023.3313934](#).
- [51] S. S. Mukherjee, C. Weaver, J. Emer, S. K. Reinhardt, and T. Austin, "A systematic methodology to compute the architectural vulnerability factors for a high-performance microprocessor," in *Proc. IEEE/ACM 36th Annu. Int. Symp. Microarchitecture*, 2003, pp. 29–40, doi: [10.1109/MICRO.2003.1253181](#).

- [52] V. Sridharan and D. R. Kaeli, "Eliminating microarchitectural dependency from architectural vulnerability," in *Proc. IEEE 15th Int. Symp. High Perform. Comput. Archit.*, 2009, pp. 117–128, doi: [10.1109/HPCA.2009.4798243](https://doi.org/10.1109/HPCA.2009.4798243).
- [53] T. Tsai, S. K. S. Hari, M. Sullivan, O. Villa, and S. W. Keckler, "NVBitFI: Dynamic fault injection for GPUs," in *Proc. 51st Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw.*, 2021, pp. 284–291, doi: [10.1109/DSN48987.2021.00041](https://doi.org/10.1109/DSN48987.2021.00041).
- [54] M. Vallero, "Qufi-pennylane," Aug. 2023. [Online]. Available: <https://github.com/MarzioVallero/qufi>
- [55] D. A. G. Goncalves de Oliveira, L. L. Pilla, T. Santini, and P. Rech, "Evaluation and mitigation of radiation-induced soft errors in graphics processing units," *IEEE Trans. Comput.*, vol. 65, no. 3, pp. 791–804, Mar. 2016, doi: [10.1109/TC.2015.2444855](https://doi.org/10.1109/TC.2015.2444855).
- [56] A. Li, S. Stein, S. Krishnamoorthy, and J. Ang, "QASMBench: A low-level quantum benchmark suite for NISQ evaluation and simulation," *ACM Trans. Quantum Comput.*, vol. 4, no. 2, Feb. 2023, Art. no. 10, doi: [10.1145/3550488](https://doi.org/10.1145/3550488).
- [57] X. Fu et al., "Quingo: A programming framework for heterogeneous quantum-classical computing with NISQ features," *ACM Trans. Quantum Comput.*, vol. 2, no. 4, Dec. 2021, Art. no. 19, doi: [10.1145/3483528](https://doi.org/10.1145/3483528).
- [58] A. Suau, G. Staffelbach, and A. Todri-Sanial, "Qprof: A gprof-inspired quantum profiler," *ACM Trans. Quantum Comput.*, vol. 4, no. 1, Oct. 2022, Art. no. 4, doi: [10.1145/3529398](https://doi.org/10.1145/3529398).
- [59] D. Cuomo et al., "Optimized compiler for distributed quantum computing," *ACM Trans. Quantum Comput.*, vol. 4, no. 2, Feb. 2023, Art. no. 15, doi: [10.1145/3579367](https://doi.org/10.1145/3579367).
- [60] K. N. Smith et al., "Exploiting slack to mitigate decoherence in quantum circuits," *ACM Trans. Quantum Comput.*, vol. 4, no. 1, Oct. 2022, Art. no. 8, doi: [10.1145/3548778](https://doi.org/10.1145/3548778).
- [61] N. Casciola, E. Giusto, E. Dri, D. Oliveira, P. Rech, and B. Montrucchio, "Understanding the impact of cutting in quantum circuits reliability to transient faults," in *Proc. IEEE 28th Int. Symp. On-Line Testing Robust System Des.*, 2022, pp. 1–7, doi: [10.1109/IOLTS56730.2022.9897308](https://doi.org/10.1109/IOLTS56730.2022.9897308).



Marzio Vallero received the B.Sc. and M.Sc. degrees in computer engineering from the Politecnico di Torino, Torino, Italy, in 2020 and 2022, respectively. He is currently working toward the Ph.D. degree in information engineering and computer science with the University of Trento, Trento, Italy.

He is also enrolled in the Doctoral Transdisciplinary Program in quantum science and technologies with the University of Trento. His current research interests include distributed quantum circuit simulation, quantum error mitigation, and the reliability of quantum devices.



Emanuele Dri received the master's degree in data science and engineering in 2021, with a thesis on machine learning for text classification from the Politecnico di Torino, Torino, Italy, where he has been working toward the Ph.D. degree in quantum computing.

His research interests include the reliability of quantum circuits with respect to transient faults and developing and adapting quantum algorithms for the finance sector, helping in bridging the gap between academic research and industry.



Edoardo Giusto (Member IEEE) received the M.S. degree in computer engineering and the Ph.D. degree in computer and systems engineering from the Politecnico di Torino, Torino, Italy, in 2017 and 2021, respectively.

He is currently an Assistant Professor with the Department of Electrical Engineering and Information Technology, University of Naples Federico II, Naples, Italy. He was a Visiting Postdoc with the Superconducting Quantum Materials and Systems Center, Fermilab, Batavia, IL, USA. This postdoctoral position was part of the Next Generation Internet Transatlantic Fellowship Program, funded by the European Commission under Horizon Europe. His research interests revolve around quantum computing, encompassing applications of QC, problem mapping, reliability and fault tolerance of QC devices, and the integration of QC in high-performance computing infrastructures.

Dr. Giusto actively contributes to the field as a Technical Committee Member for the IEEE QCE - Quantum Week conference and the IEEE CTSoc Quantum in Consumer Technology. He is a Member of ACM.



Bartolomeo Montrucchio received the M.Sc. degree in electronic engineering and the Ph.D. degree in computer engineering from the Politecnico di Torino, Turin, Italy, in 1998, and 2002, respectively.

He is currently a Full Professor of computer engineering with the Dipartimento di Automatica e Informatica, Politecnico di Torino. His current research interests include image analysis and synthesis techniques, scientific visualization, sensor networks, RFIDs, and quantum computing.



Paolo Rech received the master's and Ph.D. degrees in information engineering from Padova University, Padova, Italy, in 2006 and 2009, respectively.

Since 2012, he is an Associate Professor with UFRGS, Brazil, and since 2022, he is an Associate Professor with the Università di Trento, Trento, Italy. His main research interests include the evaluation and mitigation of radiation-induced effects in autonomous vehicles for automotive applications and space exploration, in large-scale HPC centers, and quantum computers.

Dr. Rech is the 2019 Rosen Scholar Fellow with the Los Alamos National Laboratory. He was the recipient of the 2020 impact in Society Award from the Rutherford Appleton Laboratory, U.K., and the Marie Curie Fellowship from the European Commission in 2020.

Open Access funding provided by 'Università degli Studi di Trento' within the CRUI CARE Agreement