IEEE Transactions on Quantum Engineering

Received 7 January 2026; revised 23 April 2026; accepted 4 May 2026; date of publication 7 May 2026; date of current version 10 June 2026.

Digital Object Identifier 10.1109/TQE.2026.3691176

Quantum Circuit-Based Adaptation for Credit Risk Analysis

HALIMA GIOVANNA AHMAD[1] (Member, IEEE), ALESSANDRO SARNO[1], MEHDI EL BAKRAOUI[2], CARLO COSENZA[1], CLÉMENT BÉSOIN[2], FRANCESCA CIBRARIO[3], VALERIA ZAFFARONI[3], GIACOMO RANIERI[3], ROBERTO BERTILONE[3], VIVIANA STASINO[1], PASQUALE MASTROVITO[1], FRANCESCO TAFURI[1], DAVIDE MASSAROTTI[4], LEONARDO CHABBRA[2], AND DAVIDE CORBELLETTO[3]

[1]Dipartimento di Fisica "Ettore Pancini", Università degli Studi di Napoli "Federico II", 80126 Napoli, Italy
[2]G2Q Computing, 20154 Milano, Italy
[3]Intesa Sanpaolo, 10121 Torino, Italy
[4]Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione, Università degli Studi di Napoli Federico II, 80125 Napoli, Italy

Corresponding author: Halima Giovanna Ahmad (e-mail: halimagiovanna.ahmad@unina.it).

The work was supported in part by the the National Recovery and Resilience Plan (PNRR) Ministero dell'Universitá e della Ricerca under Grant CN00000013-ICSC and Grant PE0000023-NQSTI, in part by the Pathfinder EIC 2023 project "FERROMON-Ferrotransmons and Ferrogatemons for Scalable Superconducting Quantum Computers," and in part by the Project PRIN 2022-Advanced Control and Readout of scalable Superconducting NISQ Architectures (SuperNISQ) under Grant CUP E53D23001910006.

This article has supplementary downloadable material available at https://doi.org/10.1109/TQE.2026.3691176, provided by the authors.

ABSTRACT Noisy and intermediate-scale quantum (NISQ) processors are sensitive to noise, prone to quantum decoherence, and are not yet capable of continuous quantum error correction for fault-tolerant quantum computation. Hence, quantum algorithms designed in the prefault-tolerant era cannot neglect the noisy nature of the hardware, and investigating the relationship between quantum hardware performance and the output of quantum algorithms is essential. In this work, we experimentally study how hardware-aware variational quantum circuits on a superconducting quantum processing unit can model distributions relevant to specific use-case applications for credit risk analysis, e.g., standard Gaussian distributions for latent factor loading in the Gaussian Conditional-Independence model. We use a transpilation technique tailored to the specific quantum hardware topology, which minimizes gate depth and connectivity violations, and we calibrate the gate rotations of the circuit to achieve an optimized output from quantum algorithms. Our results demonstrate the viability of quantum adaptation on a small-scale, proof-of-concept model inspired by financial applications and offer a good starting point for understanding the practical use of NISQ devices.

INDEX TERMS Experimental quantum computing, quantum circuit transpilation, quantum finance, Superconducting quantum hardware.

I. INTRODUCTION

Quantum computing has been promised as a reliable solution for use cases that are inefficiently addressed by standard classical computing. Among them, financial applications have attracted considerable attention lately. In the credit risk analysis (CRA) domain, quantum strategies [1], [2], [3] for computing quantities related to the estimation of economic capital, such as the value at risk (VaR) and the conditional VaR (CVaR), rely on quantum amplitude estimation (QAE). This technique promises a theoretical quadratic speed-up with respect to its classical counterpart, the Monte Carlo simulation approach [4]. However, the reduced number of addressable qubits and the relatively high level of gate errors and decoherence in state-of-the-art quantum processing units (QPUs) represent a technological challenge to overcome for the demonstration of quantum algorithms, at least as proof of concepts [5]. Nevertheless, great advances in hardware performance enable us to explore several use cases with remarkable results already in the noisy and intermediate-scale quantum (NISQ) era [5]. Especially relevant in this framework

are superconducting QPUs (sQPUs), which have attracted noticeable interest worldwide for their remarkable circuit analogies with CMOS-based classical processors, sharing similar historical bottlenecks, advantages, limitations, perspectives, and compatibilities with state-of-the-art classical computing techniques [6]. Remarkably, superconducting technology provides extensive engineering freedom, which allows for an enriching day-by-day pool of applications in the quantum technologies realm [7], [8] and a strong variety of hardware platforms for quantum computing, each designed to tackle specific limitations of current quantum circuit designs [9]. sQPUs leverage the use of Josephson junctions to artificially engineer macroscopic artificial atoms in a way that allows easy preparation, manipulation, and readout of quantum states with an increasing level of complexity [10], [11], [12], and eventually exploit the fundamental computational resources of quantum computing: superposition and entanglement. Most importantly, the practical interface between sQPUs and classical computing platforms, e.g., cloud computing and high-performance computing, has provoked a strong speed-up and quantum awareness in the field of quantum algorithms [13], [14]. Although pushing coherence times well above current limits and inventing smart ways to overcome gate and decoherence errors [15], [16], [17] is probably the most sound solution to reach the fault-tolerant quantum (FTQ) regime [18], it is still worth systematically and coherently investigating the possible outcomes achievable in NISQ devices.

In this work, we exploit a transmon-based sQPU, the core of the Italian superconducting quantum computing center Partenope, conceived to experimentally address quantum algorithms directly at the hardware (or pulse) level. We report here the first experimental investigation and implementation of one subcircuit in the QAE-based CRA algorithm on this quantum machine. In this quantum procedure, the first part of the Grover operator's state preparation is dedicated to loading an uncertainty model that represents the probability of default (PD) of the counterparties under analysis. To perform preliminary experiments on quantum hardware, we decided to focus on this component since it remains unchanged when estimating both the VaR and the CVaR through repeated applications of the Grover operator. Moreover, the construction of the uncertainty model requires the loading of standard normal distributions $\mathcal{N}(0, 1)$, with a mean equal to 0 and a standard deviation of 1, a task that is also reusable for other quantum finance use cases, such as option pricing [19], [20], [21], [22]. Concerning the execution of the uncertainty model, the transpilation, i.e., the process of converting a quantum circuit from one representation to another (often to optimize it for a specific quantum hardware platform) plays a key role, especially in the NISQ era and in QPUs characterized by limited connectivity (as the sQPU analyzed in this work). Akin to compilation in classical computing, transpilation adapts quantum circuits for the unique characteristics and limitations of different quantum computers instead of transforming code for different processors. Although in the long-term quantum computing should be hardware agnostic, it is still not the case for NISQ devices. This work, in fact, aims to bridge a fundamental gap between quantum algorithm specialists and quantum hardware engineers, discussing in detail at the machine level the role of 1) hardware circuit parameters; 2) qubit connectivity; 3) hardware performance; and 4) electronics implementation of quantum logic gates on the actual output of a quantum algorithm, and the transpilation process that allows optimizing the outcome of a quantum circuit.

The rest of this article is organized as follows. Section II defines how the uncertainty model is mapped to a quantum circuit. We present the scalable, parameterized circuit for a single-counterparty model, whose PD is influenced by a single risk-factor, and describe how each component of the circuit was developed. Section III details the hardware we used for the executions—qubit topology and gate fidelities—and shows the transpiled circuit that reflects the device's coupling map, which differs from the idealized layout. This allows us to detail the connection between the simulated transpilation of the quantum circuits and the effective transpilation required by the machine to achieve optimal results. Section IV presents results and a comparison of the quantum implementation with a classical baseline for the same model. In Section V, we discuss on the scientific significance of this work and main advantages. Finally, Section VI concludes this article and discusses the limitations and future perspectives.

II. METHODS

A. CONTEXT OF CREDIT RISK ANALYSIS AND GAUSSIAN CONDITIONAL INDEPENDENCE MODEL

CRA quantifies the likelihood and impact of borrower default at both single-exposure and portfolio levels. The core ingredients are the PD and the loss given default (LGD), which combine to determine expected and tail losses over a horizon. Since debtors are not independent, portfolio models introduce latent risk factors to capture dependence (e.g., one-factor Vasicek or multifactor structures, often paired with copulas for joint tails). The accurate simulation of correlated default scenarios and loss distributions is fundamental for practical tasks, e.g., pricing and capital allocation, as well as Basel stress testing. The principal computational bottleneck is the generation of high-fidelity samples from these correlated models and the evaluation of portfolio losses across many scenarios.

These requirements naturally favor latent-factor formulations that balance realism with tractability. The Gaussian conditional-independence (GCI) uncertainty model [23] provides an analytically convenient link from shared risk factors to debtor-level default probabilities and is widely used as a classical baseline [24], [25]. Classically, GCI posits that the debtor-level conditional default probabilities are conditioned

on one (or several) latent normal risk factor(s) z as

$$\mathrm{PD}_k(z) = \Phi\left(\frac{\Phi^{-1}(p_k^0) - \sqrt{\rho_k}z}{\sqrt{1-\rho_k}}\right) \tag{1}$$

where Φ is the standard normal cumulative distribution function (CDF), and p_k^0 and ρ_k are, respectively, the baseline default level and the correlation of the kth debtor (asset).

For the quantum mapping of this uncertainty model, we define the conditional default probability $\mathrm{PD}_k(z)$ as the probability of measuring the state $|1\rangle$ on a target qubit

$$\mathrm{PD}_k(z) \equiv P_1 = \sin^2(\alpha_k z + \beta_k)\,. \tag{2}$$

Here, the parameters α_k and β_k are obtained by representing the classical conditional default probability function in a linear form within the latent factor z, such that the corresponding quantum rotation angle depends affinely on z with a slope α_k and an offset β_k. Such parameters have been computed analytically, and a detailed explanation on the calculation is available in Supplementary Material [Formula (5) for β_k and (6) for α_k of Supplementary Material]. In this way, conditional default mechanisms can be implemented by parameterizing single-qubit rotations of the type

$$R_y\left(2(\alpha_k z + \beta_k)\right) \tag{3}$$

so that the amplitude of $|1\rangle$ encodes the desired conditional default probability.

In the circuit, the latent factor z is represented by an n-qubit register, whose computational basis states encode a discrete set of latent factor values. Absorbing these discrete values into effective linear parameters with slope $\tilde{\alpha}_k$ and offset $\tilde{\beta}_k$, the gate implemented on the target qubit can be written compactly as

$$R_y\left(2(\tilde{\alpha}_k\,\hat{z}_{\mathrm{code}} + \tilde{\beta}_k)\right) \tag{4}$$

where $\hat{z}_{\mathrm{code}}$ denotes the integer value encoded by the n-qubit z-register and controls the rotation angle, thereby encoding the conditional default probabilities $\mathrm{PD}_k(z)$ across the encoded configurations (see Supplementary Material for further details).

This setting is typical of a single risk factor model, but in general the modeling requests the encoding of more of them. The total number of qubits required for encoding the GCI model is, in fact, determined by two contributions: 1) the encoding of assets default event and 2) the representation of the risk factors. In the present setting, each asset default event is encoded with one qubit, so the number of qubits related to this purpose scales linearly with the number of assets in the portfolio. In addition, each risk factor requires a dedicated register to load its corresponding normal distribution. The size of this register depends on the chosen discretization, which plays a crucial role in the accuracy of the model: allocating more qubits to a risk-factor register allows for a finer discretization, and therefore for a larger number of bins to represent the corresponding distribution. As a result, the total qubit cost depends not only on the number of assets and

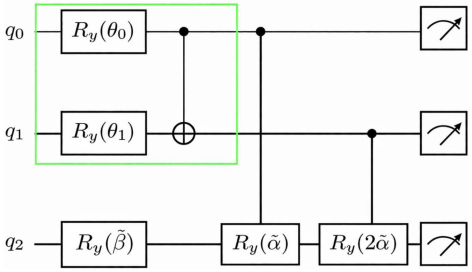


FIGURE 1. Quantum circuit for the target GCI model with one asset and one risk factor. The green box underlines the Gaussian loading subcircuit.

risk factors but also on the level of discretization used for each risk factor, as

$$N_{\mathrm{qubits}} = N_{\mathrm{assets}} + \sum_{k=1}^{N_{\mathrm{risk}}} n_k \tag{5}$$

where N_{assets} is the number of qubits used to encode the assets, N_{risk} is the number of risk factors, and n_k is the number of qubits in the register used to discretize the kth risk factor. In this work, we focused on a scenario involving one asset and one latent factor employing three qubits: q_0 and q_1 form the z-register ($n = 2$) that prepares a discrete normal distribution over the latent factor z, while q_2 is the asset qubit that encodes the conditional default probability. As an example, scaling the model to two assets and two risk factors requires at least 6 qubits. Two qubits are used to represent the assets default event, while each risk factor is encoded with $n_k = 2$ qubits, allowing every discretized normal distribution to be represented with $2^{n_k} = 4$ bins.

In real credit-risk applications, portfolio size can range from hundreds to many thousands of exposures, while the number of systematic risk factors depends on the modeling framework. This shows directly that moving from toy models to real business cases can quickly increase the hardware requirements. For example, a portfolio with 1000 assets already requires 1000 qubits for the assets' default encoding alone, and adding ten risk factors, discretized with 6 qubits each, increases the total to 1060 qubits.

In the one-risk-factor/one-asset scenario, the loading of the latent factor uses two single-qubit R_y rotations followed by a $\mathrm{CNOT}_{0,1}$ entangler, shaping the two-qubit amplitudes to approximate a Gaussian over the (q_0, q_1) register (the green square in Fig. 1). The state of the asset (default) qubit q_2 can be prepared via controlled R_y rotations conditioned on q_0 and q_1, such that the effective rotation angle, and consequently the probability of measuring $|1\rangle$ on q_2, depends on the encoded value of the normal factor. This realizes the conditional map $\mathrm{PD}(z) \approx \sin^2(\alpha z + \beta)$, linking the risk factor to the asset's default probability within a shallow, hardware-efficient circuit.

B. VARIATIONAL CIRCUIT FOR A NORMAL DISTRIBUTION

A crucial role in the CRA and GCI models is played by the preparation of a discrete normal distribution over the

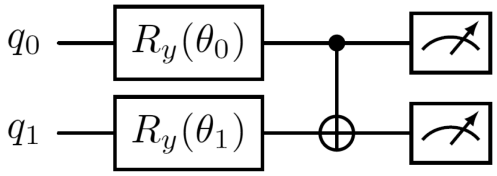


FIGURE 2. Two-qubit quantum circuit for the Gaussian distribution loading.

latent factor. In fact, the GCI quantum circuit in Fig. 1 includes a Gaussian loading subcircuit on a register of 2 qubits (green box). The generalization to n qubits, spanning 2^n computational basis states, yields the requirement to prepare a quantum state $\psi(\boldsymbol{\theta})$. According to [26],

$$|\psi(\boldsymbol{\theta})\rangle = \sum_{b=0}^{2^n-1} \sqrt{p_b(\boldsymbol{\theta})}\,|b\rangle \tag{6}$$

so that measuring in the computational basis returns the bitstring b with a probability of $p_b(\boldsymbol{\theta})$.

We first adopt a distribution-agnostic strategy: select a hardware-efficient ansatz and train its parameters $\boldsymbol{\theta}$ so that the model distribution $p(\boldsymbol{\theta})$ matches a prescribed histogram

$$p^\star = \{p_b^\star\}_{b=0}^{2^n-1}. \tag{7}$$

Given a discrepancy functional $D(p(\boldsymbol{\theta}) \,\|\, p^\star)$, gradients are estimated via the parameter–shift rule, and $\boldsymbol{\theta}$ is updated with a stochastic optimizer, such as Adam [27], [28], [29], which is preferable compared to other optimizers for variational quantum algorithms [30], since it uses more direct and stable gradient-based updates, leading to faster and smoother convergence when the optimization landscape is simple.

This recipe—compact ansatz, target histogram, shift-rule gradients, and Adam updates—supports arbitrary discrete distributions over the register.

As concrete instances, we synthesize discrete Gaussians on two and three qubits.

Two Qubits: On the register (q_0, q_1), we use the hardware-efficient ansatz: single-qubit R_y rotations on each qubit followed by one entangler $\mathrm{CNOT}_{0,1}$ (Fig. 2). Let $\boldsymbol{\theta}$ collect all rotation angles; the Born probabilities are

$$p_b(\boldsymbol{\theta}) = \left|\langle b \mid \psi(\boldsymbol{\theta})\rangle\right|^2, \qquad b \in \{0, 1, 2, 3\}. \tag{8}$$

Three Qubits: We extend the circuit to (q_0, q_1, q_2) by adding an $R_y(\theta_2)$ on q_2 and a second entangler $\mathrm{CNOT}_{0,2}$ (control q_0) (Fig. 3). The output probabilities are

$$p_b(\boldsymbol{\theta}) = \left|\langle b \mid \psi(\boldsymbol{\theta})\rangle\right|^2, \qquad b \in \{0, 1, \ldots, 7\}. \tag{9}$$

Target and Training: For either $n \in \{2, 3\}$ qubits, we map each basis index b to a grid point $z(b) \in [-z_{\max}, z_{\max}]$ via an affine transform and define the target histogram of $\mathcal{N}(\mu, \sigma^2)$ as

$$p_b^\star \propto \exp\left(-\frac{(z(b)-\mu)^2}{2\sigma^2}\right), \qquad b = 0, \ldots, 2^n - 1. \tag{10}$$

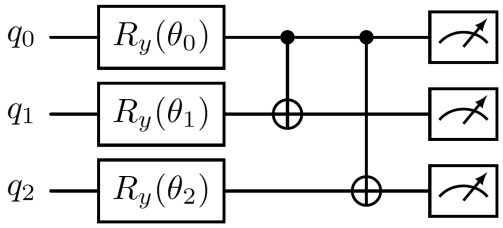


FIGURE 3. Three-qubit quantum circuit for the Gaussian distribution loading.

Parameters are learned by minimizing the distributional loss

$$\mathcal{L}(\boldsymbol{\theta}) = \sum_{b=0}^{2^n-1} \left(p_b(\boldsymbol{\theta}) - p_b^\star\right)^2 \tag{11}$$

using parameter-shift gradients and Adam updates.

After convergence, the measured histograms for the two- and three-qubit circuits closely match the target normal specified by (μ, σ), yielding shallow, hardware-efficient loaders suitable for subsequent calibration and hyperparameter tuning on the sQPU.

C. HARDWARE-LEVEL NORMAL DISTRIBUTION PREPARATION: TWO AND THREE QUBITS

Starting with a two-qubit system, we study how the gates affect the probability amplitudes of the basis $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$. The single-qubit rotation

$$R_Y(\theta) = \begin{bmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{bmatrix} \tag{12}$$

acts on each qubit. From the initial state $|00\rangle$, whose amplitude vector is $[1, 0, 0, 0]^\top$, we obtain

$$\begin{matrix} |00\rangle \\ |01\rangle \\ |10\rangle \\ |11\rangle \end{matrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \xrightarrow{R_Y(\theta_0)\otimes R_Y(\theta_1)} \begin{bmatrix} \cos(\theta_0/2)\cos(\theta_1/2) \\ \cos(\theta_0/2)\sin(\theta_1/2) \\ \sin(\theta_0/2)\cos(\theta_1/2) \\ \sin(\theta_0/2)\sin(\theta_1/2) \end{bmatrix}.$$

Applying a CNOT with control q_0 and target q_1 yields

$$\begin{bmatrix} \cos(\theta_0/2)\cos(\theta_1/2) \\ \cos(\theta_0/2)\sin(\theta_1/2) \\ \sin(\theta_0/2)\sin(\theta_1/2) \\ \sin(\theta_0/2)\cos(\theta_1/2) \end{bmatrix} \tag{13}$$

i.e., the amplitudes of $|10\rangle$ and $|11\rangle$ are swapped. Consequently, the output probabilities depend on the choices of θ_0 and θ_1.

For a symmetric, bell-shaped target (e.g., a discrete Gaussian), we require

$$P_{|00\rangle} = P_{|11\rangle}, \qquad P_{|01\rangle} = P_{|10\rangle}, \qquad P_{|00\rangle,|11\rangle} < P_{|01\rangle,|10\rangle}.$$

From (13), these symmetries are satisfied when

$$\theta_0 = \pm\frac{(2n_0+1)\pi}{2}$$
$$\theta_1 = 2\pi n_1 \pm \frac{(2n_0+1)\pi}{2} n_0, n_1 \in \mathbb{N} \tag{14}$$

and the "central-mass" condition $P_{|00\rangle,|11\rangle} < P_{|01\rangle,|10\rangle}$ further imposes

$$\cos(\theta_1/2) < \sin(\theta_1/2) \iff \pi/2 < \theta_1 < 3\pi/2 \pmod{2\pi}. \tag{15}$$

By taking the software-optimized loader as our baseline, we can then fix θ_0 as the reference angle and tune θ_1 to match the characteristics of the quantum hardware at hand. The output profile is uniform when $\theta_1 = \{\pi/2,\ 5\pi/2, \ldots\}$, while to obtain a Gaussian-like shape, one can choose θ_1 so that $\cos(\theta_1/2) < \sin(\theta_1/2)$, which yields $P_{|00\rangle,|11\rangle} < P_{|01\rangle,|10\rangle}$.

Extension to Three Qubits: We now scale to a three-qubit register with basis $\{|000\rangle, |001\rangle, |010\rangle, |011\rangle, |100\rangle, |101\rangle, |110\rangle, |111\rangle\}$. Starting from the two-qubit output (with $\theta_0 = \pi/2$) and adding an $R_Y(\theta_2)$ on q_2, then a second entangler $\mathrm{CNOT}_{0,2}$ (control q_0, target q_2), the amplitude trajectory reads

$$\frac{1}{\sqrt{2}}\begin{bmatrix}\cos(\theta_1/2)\\0\\\sin(\theta_1/2)\\0\\\sin(\theta_1/2)\\0\\\cos(\theta_1/2)\\0\end{bmatrix} \xrightarrow{R_Y(\theta_2)} \frac{1}{\sqrt{2}}\begin{bmatrix}\cos(\theta_1/2)\cos(\theta_2/2)\\\cos(\theta_1/2)\sin(\theta_2/2)\\\sin(\theta_1/2)\cos(\theta_2/2)\\\sin(\theta_1/2)\sin(\theta_2/2)\\\sin(\theta_1/2)\cos(\theta_2/2)\\\sin(\theta_1/2)\sin(\theta_2/2)\\\cos(\theta_1/2)\cos(\theta_2/2)\\\cos(\theta_1/2)\sin(\theta_2/2)\end{bmatrix}$$

$$\xrightarrow{\mathrm{CNOT}_{0,2}} \frac{1}{\sqrt{2}}\begin{bmatrix}\cos(\theta_1/2)\cos(\theta_2/2)\\\cos(\theta_1/2)\sin(\theta_2/2)\\\sin(\theta_1/2)\cos(\theta_2/2)\\\sin(\theta_1/2)\sin(\theta_2/2)\\\sin(\theta_1/2)\sin(\theta_2/2)\\\sin(\theta_1/2)\cos(\theta_2/2)\\\cos(\theta_1/2)\sin(\theta_2/2)\\\cos(\theta_1/2)\cos(\theta_2/2)\end{bmatrix}.$$

The combined action of $\mathrm{CNOT}_{0,1}$ and $\mathrm{CNOT}_{0,2}$ (control q_0 in both cases) swaps the pairs $|100\rangle \leftrightarrow |101\rangle$ and $|110\rangle \leftrightarrow |111\rangle$. Under the same symmetry conditions on θ_0 and θ_1 as above, a symmetric three-qubit profile requires

$$P_{|000\rangle} = P_{|111\rangle} \quad P_{|001\rangle} = P_{|110\rangle} \quad P_{|010\rangle} = P_{|101\rangle}$$

and

$$P_{|011\rangle} = P_{|100\rangle}$$

which is obtained when

$$\theta_2 = 2\pi n_2 \pm \theta_1, \qquad n_2 \in \mathbb{N}. \tag{16}$$

To enforce a central tendency (highest mass near $|000\rangle$ and $|111\rangle$) with strictly decreasing "rings",

$$P_{|000\rangle,|111\rangle} < P_{|001\rangle,|110\rangle} < P_{|010\rangle,|101\rangle} < P_{|011\rangle,|100\rangle}$$

one further selects θ_2 so that $\cos(\theta_2/2) < \sin(\theta_2/2)$, i.e.,

$$\pi/2 < \theta_2 < 5\pi/2 \pmod{4\pi}. \tag{17}$$

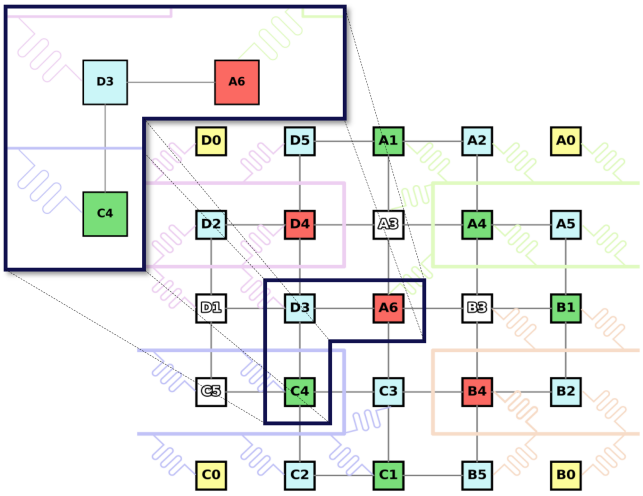


FIGURE 4. Schematics of the processor, with focus on three-qubit register A6, D3, and C4. The processor has four feedlines for multiplexed readout: feedline A (light green), coupled to 6 qubits, and feedlines B (pink), C (indigo), and D (purple), each coupled to 5 qubits. (Green) Low-frequency qubits . (Blue) Mid-frequency qubits . (Red) High-frequency qubits . (Yellow) Isolated qubits. (White) White color identifies Qubits that are not operational.

These analytic angle relations provide hardware-level guidance for shaping symmetric, bell-like histograms with shallow entangling patterns on today's sQPUs. In fact, rotation angles in sQPUs can be controlled by adjusting the amplitude and the shape of microwave pulses at room temperature with a dedicated FPGA-based electronics [31]. As we will show in Section IV, a variational optimization of such electrical parameters allows for a hardware-aware hypertuning of rotation angles, first optimized classically through quantum circuit simulations. This method eventually counteracts for systematic errors arising at the electronics level, such as amplitude and phase-errors, and automatically takes into account the intrinsic noisy nature of current NISQ processors, as well as the electrodynamics of different qubits in the same processor. Therefore, in Section IV, we examine how sQPU connectivity and circuit choices (e.g., entangler placement and single-qubit rotation settings) affect the quality of the prepared distribution.

III. HARDWARE DESCRIPTION

The sQPU used in this work is a Contralto-D from Quantware (Fig. 4), composed of 17 operational symmetric flux-tunable transmons, connected in a rectangular 2-D matrix through fixed high-frequency bus resonators.

Qubits are designed to fall into three frequency bandwidths: 1) high-frequency qubits have frequencies of the order of 6 GHz (in red); 2) medium-frequency qubits have frequencies of the order of 5 GHz (in cyan); and 3) low-frequency qubits lie in a frequency range of the order of 4 GHz (in green). All the qubits are equipped with a dedicated superconducting readout resonator, equally distributed over four readout feedlines, thereby guaranteeing multiplexed readout. They also use dedicated drive and flux lines

TABLE 1. Contralto-D QPU Specifications for A6, D3, C4 Qubits: Coherence Times (T_1 - Relaxation Time, T_2^* - Ramsey Coherence Time, T_{2E} - Hahn-Echo Coherence Time, With Errors Provided by 100 Repeated Experiments), Average Single-Qubit Gate (F_{avg}) and Readout (F_{RO}) Fidelities, and Two-Qubit CZ Interleaved Average Gate Fidelity F_{CZ} for the Pairs

	$T_1\ (\mu s)$	$T_2^*\ (\mu s)$	$T_{2E}\ (\mu s)$	$F_{\text{avg}}\ (\%)$	$F_{\text{RO}}\ (\%)$	$F_{CZ}\ (\%)$
A6	43 ± 1	37 ± 3	40 ± 1	99.855 ± 0.005	0.95 ± 0.01	
C4	38 ± 1	20 ± 1	42 ± 4	99.924 ± 0.002	0.95 ± 0.01	
D3	32 ± 1	20 ± 6	37 ± 6	99.91 ± 0.01	0.94 ± 0.01	
A6-D3						99.7 ± 0.2
D3-C4						99.05 ± 0.07

In green low frequency band qubits, in blue mid frequency band qubits and in red high frequency band qubits.

to implement single-qubit X–Y-gates and Z-gates, as well as the native two-qubit gate of the sQPU, the Conditional-Z (CZ) gate, which we implement by using sudden-net-zero flux pulses on the highest frequency qubit in a pair [32]. Flux lines also allow for parking the qubits at specific working points by using static magnetic flux. In this work, we have focused on qubits C4, D3, and A6 (low-, medium-, and high-frequency qubits, respectively), which have been operated at their flux sweet-spots. The remaining qubits of the matrix have been operated at their antisweet spot, i.e., nominally at zero frequency, to limit the effect of frequency crowding and qubit crosstalk.

The sQPU is installed at the coldest stage of a Bluefors XLD1000SL, equipped with cryogenic attenuated drive and read-in lines, superconducting cryogenic coaxial flux lines, and read-out lines with high-electron mobility transistors amplifiers, anchored at the 4 K plate of the cryostat, and a double junction isolator at the MXC. Additional detail on the cryogenic experimental setup is reported in [33, Supplementary Material]. At room temperature, the sQPU interfaces with a host PC through LAN connections via a Qblox cluster, equipped with FPGA-based modules for control, read-out, and flux pulsing of qubits. Static flux-pulses for qubits parking are delivered by low-noise DC electronics, the stable power and interface (SPI) rack, and combined with flux-pulsed signals from the cluster using bias-tees at room temperature. The Qblox electronics can be controlled by a Python-based framework, Quantify [34], which allows us to design experimental schedules to run quantum circuits of varying complexity, control every physical quantity of X, Y, Z, and decompose the required two-qubit CNOT gate in terms of the native CZ gate directly at the pulse-level. Specifically, in this sQPU, X- and Y-gates are implemented using derivative reduction adiabatic gate (DRAG) microwave pulses of 60 ns: rotation angles are set by changing the amplitude of such pulses, by using as an input the rotation angle in degree directly dependent on the amplitude value, while the rotation axis is selected by changing the phase of the microwave signals (again, setting an angle in degree, directly connected to the amount of phase-delay in the microwave signal). Microwave pulses are also used to excite the dedicated superconducting resonators, and readout is performed by digitizing the output signal through the readout modules. Meanwhile, Z single-qubit gates are virtually implemented by phase updates of the control signal [35], [36]. All other single-qubit rotations are generally decomposed in terms of the X-, Y-, and Z-rotations. For additional details see [10].

In Table 1, we report on the coherence times, gate, and read-out fidelities of the investigated qubits, acquired right before the experiments to have available benchmarking coherence and fidelity parameters of the NISQ machine. This comes in handy when we want to investigate any relation between a quantum algorithm output and the noise of the NISQ processor [33]. Additional details on calibration procedures and the gate and readout fidelities, as well as the assignment calibration matrices for the experiments reported in this work, are collected in the Supplementary Material.

IV. RESULTS

A. TWO-QUBIT NORMAL DISTRIBUTION ENCODING AND LOADING

The first experimental investigation involved the D3–C4 pair, and the goal was to identify the best θ_0, θ_1 angles combination to achieve a two-qubit standard normal distribution. By preparing the two-qubit system in the $|00\rangle$ state, we first applied an R_y gate on D3 with fixed $\theta_0 = 90^\circ$ and $\theta_1 = 111^\circ$, obtained through the target and training procedure and the Adam optimization discussed in Section II-B. The experimental result on the quantum machine is reported in Table 2. With the goal of experimentally assessing at the hardware level the effect of the rotation angle on the probability distribution shape, we swept the angle θ_1 for the R_y gate on C4 in the range $[90^\circ, 450^\circ]$ with a step of 21°. To provide a comparison with the initial experiment, in Table 2 we collect some of the acquired output distributions to show that they exhibit a periodic trend: for $90^\circ < \theta_1 < 270^\circ$, the probability amplitude distributions show a positive concave shape; for $270^\circ < \theta_1 < 450^\circ$, the concavity inverts, while for θ_1 around 90°, 270°, and 450°, the distribution becomes uniform. The experimental probability amplitudes for the two-qubit basis states as a function of all the angles θ_1 are also reported in Fig. 5, showing a reasonable agreement with the theoretical expectation. This first screening allowed us to identify a promising region with a positive concave shape and probabilities compatible with the simulated output. We repeated the experiment by thickening the step to 1° between these angles to find the optimal θ_1, which resulted in $\theta_1 =$

TABLE 2. Two-Qubit Circuit Output on D3-C4 Register as a Function of θ_1 and Fixed $\theta_0 = 90°$

	θ_0=90°				
θ_1	90°	111°	279°	300°	447°
	0.26	0.20	0.31	0.38	0.26
	0.24	0.33	0.22	0.13	0.26
	0.25	0.31	0.22	0.13	0.21
	0.25	0.16	0.25	0.37	0.27

For $\theta_1 \in \{90°, 270°\}$ probability amplitudes feature a Gaussian-like shape, while for $\theta_1 \in \{270°, 450°\}$ the concavity is inverted.

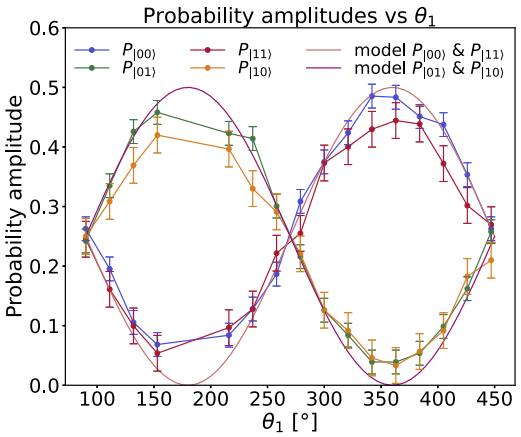


FIGURE 5. Comparison between probability amplitudes of $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$ states and theoretical model as a function of $\theta_1 \in \{90°, 450°\}$ and fixed $\theta_0 = 90°$.

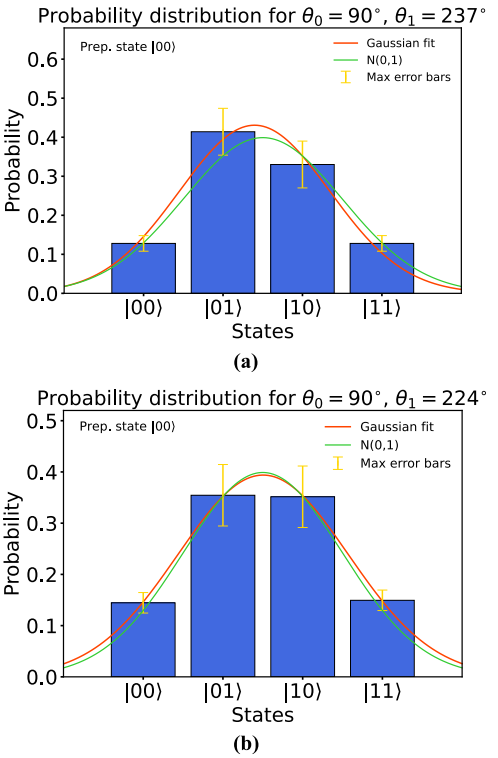


FIGURE 6. Two-qubit Gaussian distribution at the optimal θ_0 and θ_1 angle and corresponding Gaussian fit for two different pairs. In (a), experimental results for pair D3–C4. In (b), experimental results for pair D3—A6. In green, the standard normal distribution is plotted for comparison.

237° [Fig. 6(a)]. Similar experiments were also performed by preparing $|11\rangle$ as the initial state, which can be found in the Supplementary Material.

As one can notice, the quantum circuit output can reveal fluctuations that may make the output distribution slightly asymmetric. This can be accounted for state preparation and measurement (SPAM) errors in an NISQ sQPU. Therefore, we have investigated such an error by performing a statistical analysis of 100 measurements for the optimal set of angles. This provided us with a way to estimate the error bars on the quantum circuit output in terms of the population states. In detail, we assumed the deviation from the symmetric case, i.e., $P_{|00\rangle} = P_{|11\rangle}$ and $P_{|01\rangle} = P_{|10\rangle}$, so $\Delta P_{|00\rangle,|11\rangle} = P_{|00\rangle} - P_{|11\rangle}$ and $\Delta P_{|01\rangle,|10\rangle} = P_{|01\rangle} - P_{|10\rangle}$. The detailed analysis for one example output distribution can be found in the Supplementary Material.

Finally, to investigate the role of hardware connectivity in this quantum algorithm, we performed the same experiments on the pair D3 and A6, with A6 (the highest-frequency qubit in the pair) as the target qubit. As discussed in the Supplementary Material, to optimize the performance of the CNOT gate between D3 and A6 when A6 is both the flux-tuned qubit and the target one in the register, a counter phase Z-gate has been included in the CNOT decomposition to address depolarizing errors due to single-qubit phase-accumulation. The Gaussian distribution, obtained for an optimal angle $\theta_1 = 224°$ [Fig. 6(b)], differs from that obtained for D3–C4, and is more consistent with a standard normal distribution (highlighted in green). This result suggests that not only is it fundamental to optimize rotation angles on the quantum machine after prior optimization angles at the classical level but also that the optimal rotation angles depend on the physical and hardware parameters of the particular pair of qubits considered in the quantum register. This dependence cannot be fully captured by offline hardware-agnostic optimization.

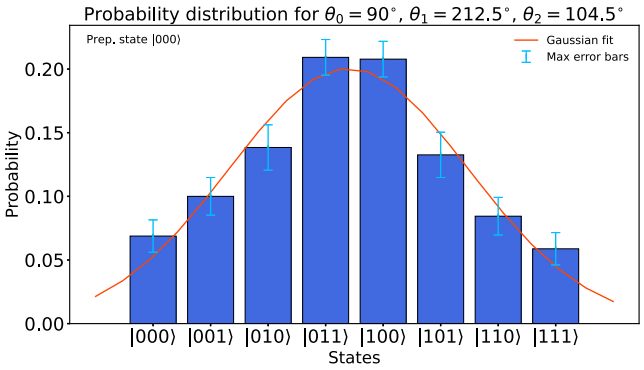


FIGURE 7. Three-qubit Gaussian distribution for qubits D3–A6–C4.

B. THREE-QUBIT NORMAL DISTRIBUTION ENCODING AND LOADING

For the three-qubit distribution sampling, we used the triplet D3–A6–C4, where the implemented quantum circuit is reported in Fig. 3. First, we fixed $\theta_0 = 90^\circ$ of the R_y gate on D3 and changed θ_1 in the range $[90^\circ, 450^\circ]$ with a step of 36° for the R_y gate on A6, and θ_2 in the range $[90^\circ, 450^\circ]$ for the R_y gate on C4. Again, the goal was to study the output distributions as a function of the angles, also obtaining a transition from positive to negative concavity, passing through a uniform output distribution (see the Supplementary Material). Then, we adjusted the step on the angles in a range for which the output distribution better satisfies the requirements of a Gaussian-like output distribution, i.e., $100^\circ < \theta_1 < 250^\circ$ with a step of 7.5°, and $90^\circ < \theta_2 < 380^\circ$ with a step of 14.5°. The optimal angle combination for the D3–A6–C4 register is $\theta_0 = 90^\circ$, $\theta_1 = 212.5^\circ$, and $\theta_2 = 104.5^\circ$ (Fig. 7). As for the two-qubit case, we estimated errors for probability amplitudes by performing a statistic of 100 measurements and by computing the differences of amplitudes between symmetric states.

The experimental evidence of our ability to fine-tune the rotation angles for up to three qubits at a hardware level to achieve a specific output distribution demonstrates that the hardware and the experimental variational approach used here promise the possibility to legitimately address a more compelling quantum circuit such as the GCI quantum circuit.

C. GCI QUANTUM CIRCUIT

Our hardware-tested GCI circuit instantiates the one-risk-factor case for one asset [$k = 1$ in (1)]: a latent $Z \sim \mathcal{N}(0, 1)$ perturbs the asset's default probability around $p_1^0 = 0.25$ with sensitivity $\rho_1 = 0.027$. The initial algorithm, introduced and shown in Fig. 1, requires three qubits and comprises a 2-qubit Gaussian sampling algorithm (green box) and one asset preparation and rotation gate, represented by a third qubit. This quantum circuit includes a controlled-Y (CY gate) that is prevented by the connectivity of the hardware. We detail the steps performed to implement the proposed quantum algorithm as follows.

Step 1: We first transpiled the circuit using Qiskit's preset pass manager, configured for hardware-aware optimization on the target backend. This pipeline performs comprehensive algebraic and structural simplifications (e.g., commutation, cancellation, and resynthesis) while respecting the backend's basis gates and coupling constraints. For both initial qubit assignment and subsequent routing, we employed the SABRE (SWAP-based bidirectional heuristic search algorithm) [38], which prioritizes mappings that reduce SWAP insertion and overall circuit depth on constrained topologies. Gate translation utilized synthesis-based methods with an approximation budget to allow slightly inexact unitary realizations in exchange for fewer two-qubit gates and shorter depths, an advantageous tradeoff for NISQ hardware where fidelity is limited by circuit length. The final GCI transpiled quantum circuit is reported in Fig. 8.

Step 2: We implemented the transpiled quantum circuit. Initially, the implementation of this quantum circuit on the machine led to unsatisfactory results, with significant deviations from the simulated output.

Step 3: We systematically simulated and experimentally implemented on the machine the transpiled circuit layer-by-layer. By systematically investigating the output of each quantum subcircuit's layer through both the Qiskit simulator and the actual hardware, the output of the subcircuit in the blue box of Fig. 8 results in a uniform three-qubit distribution. The investigation detailed in Section IV-B on the generation of a three-qubit normal distribution by exploiting the quantum circuit in Fig. 3 allowed us not only to understand for which rotation angles one can prepare a standard normal three-qubit distribution but also to identify for which angles we were able to implement a uniform three-qubit distribution. In fact, Fig. S12 of the Supplementary Material shows that there are specific angles regions where a three-qubit uniform distribution can be eventually recovered.

Step 4: We have opted for replacing the SABRE-based transpiled subcircuit in the blue box with the one in Fig. 3, achieving the configuration presented in Fig. 9. Here, the R_y rotation angles θ_0 and θ_1 (green box) control the shape of the two-qubit distribution that represents the financial variable z. This subcircuit must prepare a standard normal distribution, as previously discussed. As for the R_y rotation angle θ_2, conditioned by the angles θ_1 and θ_0, it controls on one hand the shape of the three-qubit uniform distribution that replaces the SABRE-based transpiled subcircuit in the blue box of Fig. 8. On the other hand, it controls the asset preparation, and in part accounts for the transpiled realization, directly at the hardware level, of the GCI quantum circuit, together with the remaining R_y rotation angles θ_3 and θ_4. In other words, $R_Y(\theta_2)$, $R_Y(\theta_3)$, and $R_Y(\theta_4)$ are initially derived from the offset and slope of the linearized PD in the noiseless quantum circuit model by the Adam and SABRE optimizers, but after transpilation and optimization, the circuit no longer preserves a direct correspondence between these angles and the original offset/slope parameters. Therefore, in the final hardware-ready transpiled quantum circuit, they should

IEEE Transactions on Quantum Engineering

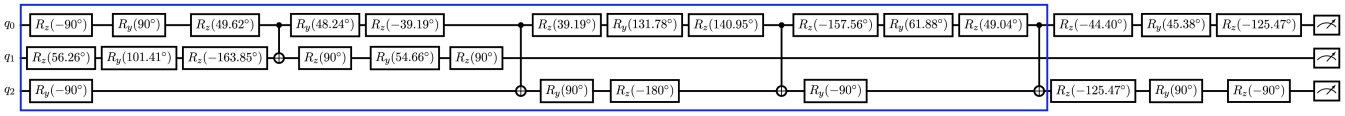


FIGURE 8. Use-case GCI transpiled quantum circuit. In the blue square, the sub circuit provides an output equivalent to a three-qubit uniform distribution circuit.

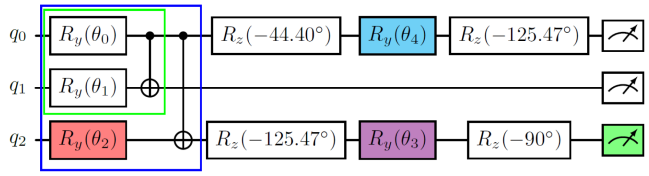


FIGURE 9. Transpiled GCI model quantum circuit. The first CNOT is here decomposed in terms of Hadamard and CZ gates, and includes a counter phase Z gate with an angle of $-135°$ to counteract CZ depolarizing errors, as discussed in Supplementary Material. The green box represents the 2-qubit normal distribution loading, while the blue box replaces the blue one in Fig. 8.

be regarded as tunable hyperparameters whose values are specific to the instance considered here.

Step 5: Having identified a reasonable transpilation of the initial GCI quantum circuit directly at the hardware level, the role of the rotation angles θ_2, θ_3, and θ_4 changes. As tunable hyperparameters, we propose here to use the pulse-level variational approach used for the optimization of the Gaussian distribution, for the search of the optimal rotation angles in the final hardware-ready transpilation of the GCI quantum circuit.

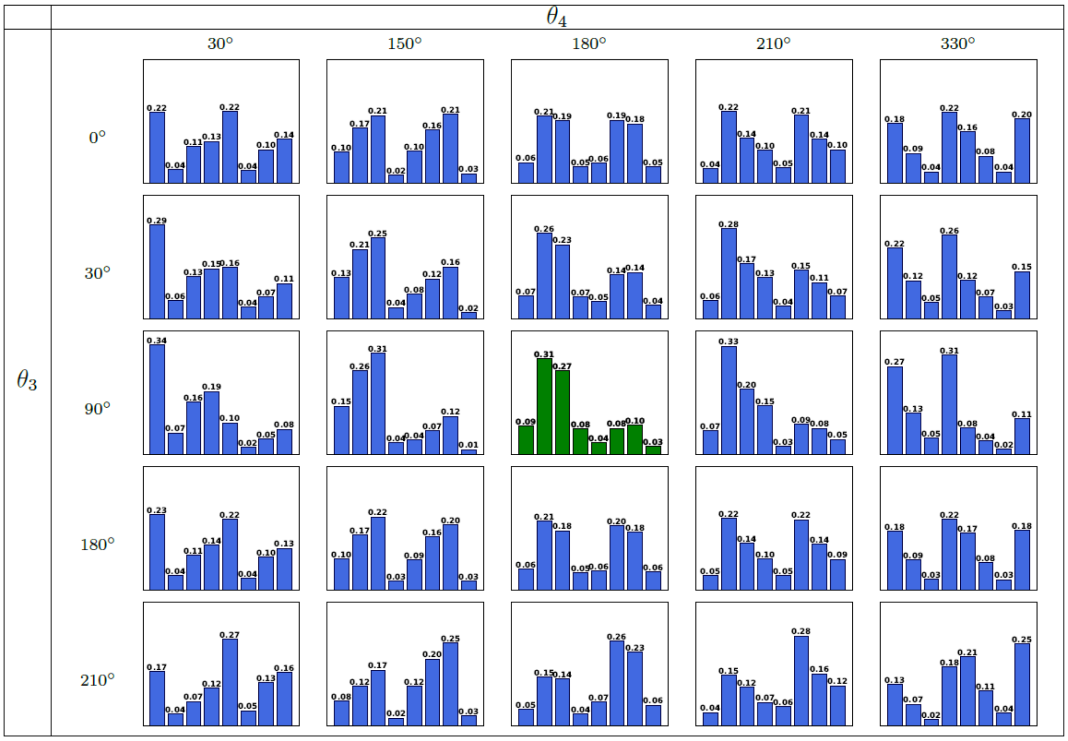
Specifically, we exploited qubits D3 and A6 to load the financial variable z as a standard normal distribution. We recall that the optimal combination of rotation angles that gives a standard normal distribution for this pair is $\theta_0 = 90°$ on D3 and $\theta_1 = 224°$ on A6. C4 plays the role of the asset. Remarkably, the rotation angle θ_2, combined with θ_0 on D3 and θ_1 on A6, allows to obtain the expected three-qubit uniform distribution at the blue box subcircuit layer. We performed experiments by changing $\theta_{3,4}$ angles in the interval $[0°, 360°]$ to adapt the algorithm efficiency to the hardware, while keeping the asset angle rotation $90°$ fixed $R_Y(\theta_2)$ (red square in Fig. 9). In Table 3, we collect the experimental output of the GCI hardware-ready transpiled quantum circuit in Fig. 9 as a function of the $\theta_{3,4}$ rotation angles.

As done for the two- and three-qubit distribution preparations, we here discuss on the dependence of the output distributions on the varied angles. First of all, the experimental output resembles a bi-modal distribution in most cases. $R_Y(\theta_3)$ (purple square in Fig. 9) affects the height of the left or right Gaussian with a periodicity of $90°$, while $R_Y(\theta_4)$ (blue square in Fig. 9) acts on the single Gaussian symmetry, with optimal behavior obtained for $R_Y(\theta_4) \in [150°, 210°]$. The profile that is most similar to the simulated output of the quantum circuit transpiled using Qiskit's preset pass manager [Fig. 10(a)] is highlighted in green and obtained for $\theta_3 = 90°$ and $\theta_4 = 180°$. As for the asset angle $R_Y(\theta_2)$ (red square in Fig. 9), in Table 2 of the Supplementary Material, we show that it is the experimental parameter that controls the symmetry between the two Gaussians at optimal $\{\theta_3, \theta_4\} = \{90°, 180°\}$.

Finally, to assess the impact of the quantum implemented GCI on a relevant outcome for credit analysis we have used these data to calculate classically the CDF as follows. The preprocessing stage decomposes each measured bit string into two logical components: 1) the leftmost bits encode default indicators and 2) the rightmost bits represent the latent variable Z. For every observed outcome with an associated probability, the rightmost bits are converted to an integer index, allowing the marginal probability $P(Z = z)$ to be accumulated. The default bits determine which debtors have defaulted; whenever a default bit is equal to one, the corresponding probability contributes to the marginal default probability, and the LGD value is added to the total loss for that scenario. After iterating over all outcomes, the procedure constructs 1) arrays of scenario losses and 2) their probabilities, from which the expected loss is computed. Identical loss values are collapsed to form a discrete probability density function (e.g., the PDF) and its CDF. In Fig. 10, we compare the CDF for the simulated transpiled quantum circuit with Qiskit's preset pass manager (Fig. 8), the simulated hardware-ready transpiled quantum circuit (Fig. 9), and the experimental implementation on the hardware of the latter. The circuits' outputs are reported in Fig. 10(a)–(c), respectively.

The hardware-ready transpilation on the machine, affected by a readout error of about $\sim 5\%$, produces a CDF [Fig. 10(c)] with a discrepancy of $\sim 0.5\%$, with respect to the simulation of the Transpiled circuit [Fig. 10(a)], indeed showing a strong agreement with the expected CDF : $P(L \leq 0) \approx 0.75$ and $P(L \leq 1000) = 1.0$. The main result is therefore not the recovery of the 95% VaR itself, since the VaR is classically determined from the reference distribution, but rather the fact that the quantum algorithm produces the same distribution from which the VaR is obtained. Specifically, the Hellinger fidelity [37] between the experimental distribution [Fig. 10(c)] and the distribution simulated using the transpiled quantum circuit in Fig. 8 [Fig. 10(a)] is $F_H = (98.9 \pm 0.3)\%$. Therefore, the quantum and classical outputs leads the same distributional description and, consequently, to the same risk conclusion for this scenario. This confirms the validity of our experimental approach, which relies entirely on the performance of the available NISQ hardware, taken into account through the variational hypertuning performed

TABLE 3. GCI Circuit Outputs as a Function of θ_3 and θ_4 At Fixed $\theta_2 = 90°$



In green, the use-case circuit output. In each plot is depicted the probability amplitude for each state in $\{|000\rangle, |001\rangle, |010\rangle, |011\rangle, |100\rangle, |101\rangle, |110\rangle, |111\rangle\}$ basis of D3–A6–C4 three-qubit register.

directly on the machine. More generally, this result highlights that transpilation, while effectively reassigning and recombining unitary rotations to address connectivity constraints and circuit depth, does not account for the sensitivity of an NISQ device to specific noise sources, even though such information is essential in the NISQ era.

V. DISCUSSION

The development of quantum algorithms for finance stems as an impactful topic, which on one hand requires to design algorithms that provide a measurable improvement compared to their classical counterpart, and on the other there is a strong requirement of compliance with current state-of-the-art quantum hardware to experimentally demonstrate their feasibility.

In our knowledge, one of the most common approach in algorithm development implicitly rely on the fact that quantum processors behave ideally without noise, or fault-tolerantly [39]. Although there is a strong effort and interest in achieving FTQ computing in the near term, with a substantial research on quantum error correction from both software and hardware point of views [40], current NISQ quantum processors are not yet capable of sustaining large-scale problems.

The development of quantum algorithms in the NISQ-era by noise-modeling current quantum machines [41], [42], [43], promises to anticipate the outcome before running on the real processor. In superconducting quantum processors, for example, one can mention bit-flip, phase-flip, and decoherence errors as the most important noise sources [44]. Although available circuit simulators include reasonable noise-models for superconducting systems, these do not necessarily apply to every quantum machine design, electrodynamics, and circuital characteristics. In order to efficiently develop quantum algorithms, it is still required a strong effort to model most, if not all, noise sources of the device, since they may not be limited to the ones aforementioned. In this work, we have, in fact, relied on off-line noiseless quantum circuit simulation for a first optimization of rotation angles, which provided us a starting point for the implementation of a GCI model on an NISQ processor.

The findings of this work have highlighted three important noise sources that should be tackled and eventually included in a noise-model: 1) readout noise; 2) phase errors; and 3) depolarizing errors. As discussed in Section IV, we investigated on the impact of fluctuations on the output states discrimination due to SPAM errors, and provided a statistical analysis to evaluate a maximum error to consider when comparing simulations and experimental results. Depolarizing errors have been addressed by including a counter phase Z gate in the CNOT implementation, as detailed in Supplementary Material. As for phase-errors related to undershoot or overshoot of control pulses used to implement R_y rotations, these have been accounted for by the variational technique deployed in this work.

An alternative approach would rely on approximate noise models by using existing circuit simulators, by counteracting

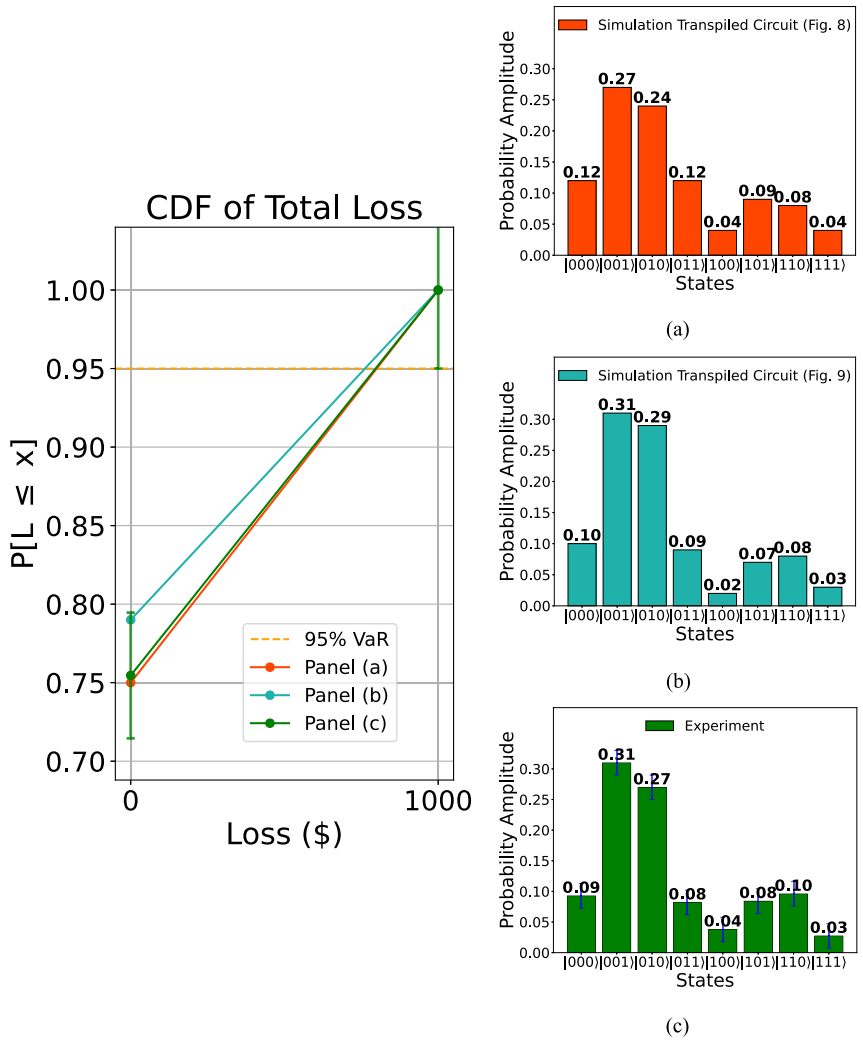


FIGURE 10. CDF of total losses for a loss $L = 1000$\$ and a VaR of 95%, calculated from the probability distribution output of the GCI model in (a) simulation of the transpiled quantum circuit in Fig. 8, in (b) simulation of the transpiled quantum circuit in Fig. 9, and in (c) experimental output of the same transpiled circuit.

remaining noise sources with error mitigation techniques. Mitigating errors in real devices is also not trivial [45]. Efficient error mitigation requires to detect and identify what noise sources are mostly impacting the machine, and consistently apply a useful mitigation strategy. Readout error mitigation, for example, allows to optimize for errors in readout state discrimination, and eventually address bit-flip errors [46]. Dynamical decoupling [47] allows to mitigate idling errors by continuously executing sequences of single-qubit operations on idle qubits that collectively function as identity gates, thereby suppressing environmental noise and relaxation. Depolarizing error mitigation, instead, involves techniques to counteract decoherence. To address depolarizing errors, common methods include Zero Noise Extrapolation [45], a randomized compiling to convert coherent errors into depolarizing noise, and estimating global depolarizing rates via partner circuits to rescale results. In principle, the identification of the best error mitigation technique relies on a systematic investigation and experimental characterization, thus not necessarily providing a more efficient solution if compared to variational approaches.

Most importantly, the study developed here provided a useful playground to identify specific noise sources that, on one hand, will be modeled and taken into account in a circuit simulator to optimize rotation angles off-line prior to the experimental realization on the hardware; on the other, the direct access to the quantum processor allowed us to devise an experimental variational approach able to counteract such effects directly in-situ, while still being able to successfully implement the GCI quantum circuit. Remarkably, the hardware-aware variational approach here proposed allowed to counteract for systematic errors on the experimental control electronics: pulsed signals phase-errors, undershoot or overshoot of pulsed signal amplitudes, delays in the control lines setup, which are not entirely compensated by calibration and optimization mechanisms of single- and two-qubit

gates rotations and would have required tailored or multiple error mitigation techniques. In a way, we exploited the implementation of proof-of-concept quantum algorithms on real NISQ devices as a tool to study noise and errors mechanisms, and to include in the optimization task the real nature of the hardware.

Although the quantum circuit-based adaptation approach applies in principle to every superconducting quantum processor, and to quantum algorithms designed for different applications, one must be aware that such a hardware-aware variational approach is not hardware-agnostic. Such an optimization depends on the processor, and in case of extended tasks involving more random variables and more qubits, as predicted in Section II, the optimization would need to be repeated for each variable in play. For example, we expect a dependence of the performance on the quantum circuit scales. Let us consider an extension from a one-risk-factor/one-asset quantum circuit to a quantum circuit that encodes two risk factors and two assets. The Qiskit's preset pass manager transpilation requires several quantum gates and qubits that roughly double with the toy-model addressed in this work. We have measured readout fidelities increasing the number of qubits from 1 to 4, and reported the readout fidelity dependence as a function of the number of qubits in Fig. S8 of Supplementary Material. The readout fidelity scales linearly with the number of qubits, and we can project to six qubits. In a three-qubit register, the readout fidelity is $\sim 88\%$, while in a six-qubit register, we expect a readout fidelity $\sim 75\%$. For these numbers, state discrimination will likely be affected, and readout error mitigation will be necessary. As for what concerns the coherence impact, and considering the single- and two-qubit gate durations reported in Supplementary Material, the total duration of the Qiskit's preset pass-manager transpiled circuit implemented on hardware was $\sim 3\,\mu s$, and has been reduced by a factor 2 in the hardware-aware transpiled circuit ($\sim 1.6\,\mu s$). For a two-risk-factor/two-asset configuration, the Qiskit's preset pass manager transpiled algorithm would require a circuit of 31 layers of gates, with an estimated total duration of $\sim 5\,\mu s$. The major contribution to the running time is due to the number of CNOT gates. Although we require experimentally implementing the algorithm to evaluate the scaling impact of the hardware-aware transpilation, it is fair to expect to achieve a reduction in time of a factor of 2 as in the one-risk-factor/one-asset scenario. By comparing such durations with the lowest coherence time of the register, i.e., $T_2^* = 20\,\mu s$, the three-qubit original transpiled circuit duration is 15% of T_2^*, and the six-qubit circuit projection should be the 25% of the coherence time. Therefore, we do not expect that coherence time is a limit for scaling to two-risk-factors/two-assets.

Finally, we have also reported a significant dependence on the qubits chosen in a register, therefore confirming the critical role played by the knowledge of the hardware to develop quantum algorithms in the NISQ era.

VI. CONCLUSION

In this work, we have provided a systematic experimental investigation of how variational quantum circuits can model distributions relevant to CRA on a superconducting quantum computer: two- and three-qubit Gaussian distributions and the GCI.

We have examined how sQPU connectivity, the different circuit nature of superconducting qubits in quantum registers, and the quantum circuit transpilation itself (e.g., entangler placement and single-qubit rotation settings) influence the desired distribution, encoded in the probability amplitudes of the prepared quantum states. A remarkable result is that the rotation angles required to realize a normal-like histogram are not generic but depend on the specific hardware characteristics of the device. We have experimentally analyzed the impact of single-qubit rotation angles on the shape of probability amplitudes distributions encoded in both two- and three-qubits registers. This allowed us not only to identify the optimal angle configuration but also, and most importantly, to determine the experimental confidence error threshold in a typical NISQ device by implementing hundreds of experimental runs.

The in-situ variational approach introduced here has also been used to optimize the GCI quantum circuit output for fixed problem parameters, taken here as an example, obtaining excellent agreement with the classical counterpart. In a sense, we have performed a hardware-aware transpilation of the quantum circuits directly at the pulse level for this use case, allowing us to capture the realistic nature of the hardware, down to the level of the individual qubits' performances and connectivities. The validity of our approach has been confirmed by the calculation of the CDF for a specific use case, obtained by using the experimental data derived from an ad-hoc transpilation of the quantum circuit, implicitly taking into account the errors of the machine, and by simulating the same quantum circuit with a noiseless simulator. Such dependence is critical in the NISQ era and emphasizes the requirement for a clear understanding of quantum computing in all its parts to achieve quantum utility in the near term: from algorithm coding to implementation through analog pulses, which allowed us to construct a more efficient quantum circuit in terms of gate depth.

Our investigation demonstrates that the readout fidelities achieved on the processor already offer satisfying results, since the Hellinger fidelities, used as a metric to quantify the deviation between the quantum and classical simulated outputs for the use-cases analyzed here, are remarkably close to the state-of-the-art values and larger than more commonly reported in literature [46], [49]. We specify that no readout error mitigation technique has been applied, nor have we integrated quantum-noise limited cryogenic amplifiers at the hardware stage. However, since we expect that the readout fidelity tends to decrease with increasing number of qubits in the registers (see Fig. 8 in Supplementary Materials), we

envision including both into our quantum algorithms implementation pipeline in the near term [33], [46], [48]. Moreover, a systematic implementation of variational quantum algorithms and circuits, not necessarily limited to the ones explored in this work, and that will eventually involve different qubits in the chip, will likely allow to uniquely define the deviation of rotation angles from the noiseless case for a fixed register. This will allow to obtain reproducible outputs on different registers of the quantum processor, once the optimized angles have been determined for each register. Moving to the envisioned generic approach in NISQ devices, the possibility to exploit knowledge on hardware errors and noise from quantum circuits implementation is the key to develop a tailored noise-model, which will be used to simulate quantum circuits, especially when a large number of qubits and parameters will be needed.

More and more research centers and industries will rely on proprietary superconducting hardware designs in the near term, with their own geometries, connectivity, characteristic circuits, and electronic parameters for quantum gate implementation. We believe this work provides a guide toward the hardware-aware implementation of other proof-of-concept quantum algorithms and motivates the use of quantum compilers that adapt circuit structure and parameterization to the specific QPU, rather than relying on a fixed, hardware-agnostic configuration.

ACKNOWLEDGMENT

Halima Giovanna Ahmad, Francesco Tafuri, and Davide Massarotti thank the SUPERQUMAP project (COST Action CA21144) and Hany Ali and Alessandro Bruno from Quantware for their support. The authors thank Johannes Hermann and Christian Junger for their support. The authors also thank Emanuele Dri from LINKS Foundation for providing useful insights in the initial stage of the project.

DATA AVAILABILITY STATEMENT

The raw data that support the findings of this study are publicly available via Zenodo Platform at https://doi.org/10.5281/zenodo.19402577.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.

REFERENCES

[1] D. J. Egger, R. Garcia Gutierrez, J. C. Mestre, and S. Woerner, "Credit risk analysis using quantum computers," *IEEE Trans. Comput.*, vol. 70, no. 12, pp. 2136–2145, Dec. 2021, doi: 10.1109/TC.2020.3038063.

[2] E. Dri et al., "A more general quantum credit risk analysis framework," *Entropy*, vol. 25, no. 4, Art. no. 593, 2023, doi: 10.3390/e25040593.

[3] D. Veronelli et al., "Implementing credit risk analysis with quantum singular value transformation," in *Proc. IEEE Int. Conf. Quantum Comput. Eng.*, 2025, pp. 2228–2237, 10.1109/QCE65121.2025.00243.

[4] A. Montanaro, "Quantum speedup of Monte Carlo methods," *Proc. Roy. Soc. A: Math., Phys. Eng. Sci.*, vol. 471, no. 2181, Sep. 2015, Art. no. 20150301, doi: 10.1098/rspa.2015.0301.

[5] S. Chen, J. Cotler, H.-Y. Huang, and J. Li, "The complexity of NISQ," *Nature Commun.*, vol. 14, no. 1, Sep. 2023, Art. no. 6001, doi: 10.1038/s41467-023-41217-6.

[6] M. H. Devoret and R. J. Schoelkopf, "Superconducting circuits for quantum information: An outlook," *Science*, vol. 339, no. 6124, pp. 1169–1174, 2013, doi: 10.1126/science.1231930.

[7] D. Salvoni et al., "Demonstration of single photon detection in amorphous molybdenum silicide / aluminium superconducting nanostrip," *IEEE Instrum. Meas. Mag.*, vol. 24, no. 5, pp. 69–74, May 2021, doi: 10.1109/MIM.2021.9491006.

[8] P. Mastrovito, H. G. Ahmad, A. Porzio, M. Esposito, D. Massarotti, and F. Tafuri, "On-chip microwave coherent source with in-situ control of the photon number distribution," *Commun. Phys.*, vol. 8, no. 1, 2025, Art. no. 295, doi: 10.1038/s42005-025-02219-6.

[9] S. Rasmussen et al., "Superconducting circuit companion—An introduction with worked examples," *PRX Quantum*, vol. 2, Dec. 2021, Art. no. 040204, doi: 10.1103/PRXQuantum.2.040204.

[10] P. Krantz, M. Kjaergaard, F. Yan, T. P. Orlando, S. Gustavsson, and W. D. Oliver, "A quantum engineer's guide to superconducting qubits," *Appl. Phys. Rev.*, vol. 6, no. 2, 2019, Art. no. 021318, doi: 10.1063/1.5089550.

[11] H. G. Ahmad et al., "Investigating the individual performances of coupled superconducting transmon qubits," *Condens. Matter*, vol. 8, no. 1, 2023, Art. no. 29, doi: 10.3390/condmat8010029.

[12] V. Stasino et al., "Implementation and readout of maximally entangled two-qubit gates quantum circuits in a superconducting quantum processor," *J. Supercond. Novel Magn.*, vol. 38, no. 2, 2025, Art. no. 129, doi: 10.1007/s10948-025-06968-x.

[13] T. Beck et al., "Integrating quantum computing resources into scientific HPC ecosystems," *Future Gener. Comput. Syst.*, vol. 161, pp. 11–25, 2024, doi: 10.1016/j.future.2024.06.058.

[14] E. Mansfield et al., "First practical experiences integrating quantum computers with HPC resources: A case study with a 20-qubit superconducting quantum computer," in *Proc. SC '25 Workshops Int. Conf. High Perform. Comput., Netwo., Storage Anal.*, Association for Computing Machinery, New York, NY, USA, 2025, pp. 1842–1850, doi: 10.1145/3731599.3767551.

[15] H. G. Ahmad et al., "Hybrid ferromagnetic transmon qubit: Circuit design, feasibility, and detection protocols for magnetic fluctuations," *Phys. Rev. B*, vol. 105, Jun. 2022, Art. no. 214522, doi: 10.1103/PhysRevB.105.214522.

[16] F. Hu et al., "Overcoming the coherence time barrier in quantum machine learning on temporal data," *Nature Commun.*, vol. 15, no. 1, Aug. 2024, Art. no. 7491, doi: 10.1038/s41467-024-51162-7.

[17] M. Tuokkola et al., "Methods to achieve near-millisecond energy relaxation and dephasing times for a superconducting transmon qubit," *Nature Commun.*, vol. 16, no. 1, Jul. 2025, Art. no. 5421, doi: 10.1038/s41467-025-61126-0.

[18] J. Preskill, "Fault-tolerant quantum computation," in *Introduction to Quantum Computation and Information*. Singapore: World Scientific, 1998, pp. 213–269, doi: 10.1142/9789812385253_0008.

[19] N. Stamatopoulos et al., "Option pricing using quantum computers," *Quantum*, vol. 4, Jul. 2020, Art. no. 291, doi: 10.22331/q-2020-07-06-291.

[20] S. Chakrabarti, R. Krishnakumar, G. Mazzola, N. Stamatopoulos, S. Woerner, and W. J. Zeng, "A threshold for quantum advantage in derivative pricing," *Quantum*, vol. 5, Jun. 2021, Art. no. 463, doi: 10.22331/q-2021-06-01-463.

[21] F. Cibrario et al., "Quantum amplitude loading for rainbow options pricing," in *Proc. IEEE Int. Conf. Quantum Comput. Eng.*, vol. 01, 2024, pp. 211–220, doi: 10.1109/QCE60285.2024.00034.

[22] F. Cibrario et al., "Autocallable options pricing with integration-based exponential amplitude loading," 2025, *arXiv:2507.19039*, doi: 10.48550/arXiv.2507.19039.

[23] M. RUTKOWSKI and S. TARCA, "Regulatory capital modeling for credit risk," *Int. J. Theor. Appl. Finance*, vol. 18, no. 05, 2015, Art. no. 1550034, doi: 10.1142/S021902491550034X.

[24] O. A. Vasicek, "The distribution of loan portfolio value," *Risk*, vol. 15, no. 12, pp. 160–162, Dec. 2002, reprinted online by Risk.net. [Online]. Available: https://www.bankofgreece.gr/MediaAttachments/Vasicek.pdf

[25] M. B. Gordy, "A risk-factor model foundation for ratings-based bank capital rules," *J. Financial Intermediation*, vol. 12, no. 3, pp. 199–232, 2003, doi: 10.1016/S1042-9573(03)00040-8.

[26] B. Paine and K. Dasgupta, "Loading probability distributions in a quantum circuit," 2022, *arXiv:2208.13372*, doi: 10.48550/arXiv.2208.13372.

[27] C. Zoufal, A. Lucchi, and S. Woerner, "Quantum generative adversarial networks for learning and loading random distributions," *Npj Quantum Inf.*, vol. 5, 2019, Art. no. 103, doi: 10.1038/s41534-019-0223-2.

[28] D.P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*, doi: 10.48550/arXiv.1412.6980.

[29] M. Ballarin, J. J. Garcıá-Ripoll, D. Hayes, and M. Lubasch, "Efficient quantum state preparation of multivariate functions using tensor networks," 2025, *arXiv:2511.15674*, doi: 10.48550/arXiv.2511.15674.

[30] A. Pellow-Jarman, I. Sinayskiy, A. Pillay, and F. Petruccione, "A comparison of various classical optimizers for a variational quantum linear solver," *Quantum Inf Process*, vol. 20, 2021, Art. no. 202, doi: 10.1007/s11128-021-03140-x.

[31] D. Ristè, S. Fallek, B. Donovan, and T. A. Ohki, "Microwave techniques for quantum computers: State-of-the-art control systems for quantum processors," *IEEE Microw. Mag.*, vol. 21, no. 8, pp. 60–71, Aug. 2020, doi: 10.1109/MMM.2020.2993477.

[32] V. Negírneac et al., "High-fidelity controlled-*Z* gate with maximal intermediate leakage operating at the speed limit in a superconducting quantum processor," *Phys. Rev. Lett.*, vol. 126, Jun. 2021, Art. no. 220502, doi: 10.1103/PhysRevLett.126.220502.

[33] H. G. Ahmad et al., "Experimental demonstration of non-local magic in a superconducting quantum processor," 2025, *arXiv:2511.15576*, doi: 10.48550/arXiv.2511.15576.

[34] Quantify, "Quantify Python framework," quantify Documentation. Accessed: Dec. 13, 2025. [Online]. Available: https://quantify-os.org/

[35] *Qblox*, "Numerically controlled oscillator," qblox Documentation 2025.10.0. Accessed: Nov. 3, 2025. [Online]. Available: https://docs.qblox.com/en/main/products/qBlox_instruments/tutorials/QRM/nco_control_adv.html

[36] D. C. McKay, C. J. Wood, S. Sheldon, J. M. Chow, and J. M. Gambetta, "Efficient *Z* gates for quantum computing," *Phys. Rev. A*, vol. 96, Aug. 2017, Art. no. 022330, doi: 10.1103/PhysRevA.96.022330.

[37] E. Hellinger, "Neue begründung der theorie quadratischer formen von unendlichvielen veränderlichen," *J. Für Die Reine Und Angewandte Mathematik*, vol. 1909, no. 136, pp. 210–271, 1909, doi: 10.1515/crll.1909.136.210.

[38] S. Niu, A. Suau, G. Staffelbach, and A. Todri-Sanial, "A hardware-aware heuristic for the qubit mapping problem in the NISQ era," *IEEE Trans. Quantum Eng.*, vol. 1, 2020, Art. no. 3101614, doi: 10.1109/TQE.2020.3026544.

[39] J. Pan, "Analyzing noise for quantum advantage," *Nature Comput. Sci.*, vol. 1, no. 12, Dec. 2021, Art. 776, doi: 10.1038/s43588-021-00178-w.

[40] E. Campbell, B. Terhal, and C. Vuillot, "Roads towards fault-tolerant universal quantum computation," *Nature*, vol. 549, pp. 172–179, 2017, doi: 10.1038/nature23460.

[41] L. Clinton, J. Bausch, and T. Cubitt, "Hamiltonian simulation algorithms for near-term quantum hardware," *Nature Commun.*, vol. 12, 2021, Art. no. 4989, doi: 10.1038/s41467-021-25196-0.

[42] S. Bordoni, A. Papaluca, P. Buttarini, A. Sopena, S. Giagu, and S. Carrazza, "Quantum noise modeling through reinforcement learning," *Quantum Sci. Technol.*, vol. 11, Nov. 2025, Art. no. 015005, doi: 10.1088/2058-9565/ae1e98.

[43] Q. Du et al., "From noise modeling to layout optimization: A framework for quantum circuit fidelity enhancement with machine learning," *Quantum Sci. Technol.*, vol. 9, no. 3, Mar. 2026, Art. no. e00464, doi: 10.1002/qute.202500464.

[44] J. Ghosh, A. G. Fowler, M. Geller, and R. Michael, "Surface code with decoherence: An analysis of three superconducting architectures," *Phys. Rev. A*, vol. 86, no. 6, Dec. 2012, Art. no. 062318, doi: 10.1103/PhysRevA.86.062318.

[45] Z. Cai et al., "Quantum error mitigation," *Rev. Mod. Phys.*, vol. 95, no. 4, Dec. 2023, Art. no. 045005, doi: 10.1103/RevModPhys.95.045005.

[46] H. G. Ahmad et al., "Mitigating errors on superconducting quantum processors through fuzzy clustering," *Adv. Quantum Technol.*, vol. 7, no. 7, 2024, Art. no. 2300400, doi: 10.1002/qute.202300400.

[47] L. Viola and S. Loyd, "Dynamical suppression of decoherence in two-state quantum systems," *Phys. Rev. A*, vol. 58, no. 4, pp. 2733–2744, Oct. 1998, doi: 10.1103/PhysRevA.58.2733.

[48] A. Y. Levochkina et al., "Investigating pump harmonics generation in a SNAIL-based traveling wave parametric amplifier," *Superconductor Sci. Technol.*, vol. 37, no. 11, 2024, Art. no. 115021, doi: 10.1088/1361-6668/ad8314.

[49] L. Schulze and J.-R. Lahmann, "Evaluating error mitigation strategies for entangled quantum states on near-term quantum computers," in *Proc. 51st Annu. Conf. German Inform. Soc.*, 2021, pp. 943–960, doi: 10.18420/informatik2021-079.

Halima Giovanna Ahmad (Member, IEEE) received the B.S. and M.S. degrees in condensed matter physics from the University of Napoli "Federico II," Naples, Italy, in 2014 and 2017, respectively, and the Ph.D. degree in physics from the University of Napoli "Federico II", in 2021.

Since 2023, she has been a Researcher and Assistant Professor of Quantum Technologies with the University of Napoli "Federico II." She is currently supervising activities in the area of her interests within the National Center for High-Performance, Big Data, and Quantum Computing. Her research interests include the experimental study of superconducting quantum devices and their applications in quantum computing.

Dr. Ahmad has also volunteered since 2019 for outreach activities as a committee board member of the IEEE Council of Superconductivity Young Professionals.

Alessandro Sarno received the bachelor's degree in physics and the master's degree in condensed matter physics, in 2021 and 2024, respectively, from the University of Naples "Federico II," Naples, Italy, where he is currently working toward the Ph.D. degree in quantum technologies, motivated by the experimental applications of quantum physics in state-of-the-art technologies.

He is currently an active member of the QCN Laboratory research group, University of Naples "Federico II," where his work focuses on studying and implementing quantum algorithms on NISQ superconducting devices at the machine level. His research interests include the interplay between hardware characteristics and experimental outcomes, with the aim of optimizing algorithmic performance under realistic noise and device constraints.

Mehdi El Bakraoui received the master's degree in mathematical physics from Mohammed V University, in 2020.

He is currently a Physicist focused on quantum algorithms, state-preparation workflows, and hardware-aware quantum circuit development. He works as a Math Analyst and Quantum Modeller with G2Q Computing, Milano, Italy, where he develops hybrid quantum–classical solvers, improves execution throughput with parallel/subprocess pipelines, and collaborates on constraint-aware transpilation under real backend coupling constraints. He also has experience in time-series forecasting and statistical modeling.

Carlo Cosenza received the master's degree in physics, in 2022, from University of Naples "Federico II," Naples, Italy, where he is currently working towards the Ph.D. degree in quantum technologies.

He is currently with the Low Temperature Quantum Technologies Lab, University of Naples "Federico II," led by Prof. F. Tafuri. He works on qubit control, interactions, and circuit-level modeling, combining theoretical understanding with numerical simulations and experimental analysis. He also works on the implementation and execution of quantum algorithms on the Partenope quantum computer, within the National Center for High-Performance, Big Data and Quantum Computing. His research interests include superconducting quantum circuits, including Josephson junctions, both conventional and unconventional, and their application to transmon qubits.

Clément Bésoin received the double master's degree in applied mathematics from École des Mines de Nancy, Nancy, France, in 2023, and in quantum engineering from CentraleSupélec, Gif-sur-Yvette, France.

He began his career as a Quantum Algorithms Developer with G2Q Computing, Milano, Italy. His work focuses on the design of hybrid quantum-classical pipelines for combinatorial optimization and machine learning problems. Thanks to his solid background in mathematics, he develops quantum-ready software that strategically combines classical techniques with emerging quantum technologies through efficient parallelization, with the aim to get a quantum advantage in real-world applications.

Francesca Cibrario received the master's degree in computer engineering from Politecnico di Torino, Torino, Italy, in 2018.

She was with the startup she co-founded during her final year as a student. She began her career with Intesa Sanpaolo, Torino, Italy. Her work has focused on innovative technologies, ranging from artificial intelligence to quantum computing. In addition to these areas, throughout her academic and professional experience, she has explored various technological domains, including mobile and backend development, as well as augmented reality applications. She is currently dedicated to researching the applicability of quantum computing in financial use cases, with particular emphasis on credit risk analysis, option pricing, and investigating how quantum computing can enhance optimization and machine learning tasks.

Valeria Zaffaroni received the master's degree in mathematics from Universitá degli Studi di Milano (UNIMI), Milano, Italy, in 2016, and the second-level master's degree in quantum communication and computing, from Politecnico di Torino, Torino, Italy, in 2023.

She is currently an IT Application Architect with the Quantum Competence Center, Intesa Sanpaolo, Torino, Italy. She is focused on quantum and quantum-inspired algorithms for optimization problems, to bridge research, innovation, and real-world applications in the financial sector. Her research interests include quantum mechanics.

Giacomo Ranieri received the bachelor's degree in informatic engineering from Universitá degli Studi di Bologna, Bologna BO, Italy, in 2014, and the master's degree in computer engineering from Politecnico di Torino, Torino TO, Italy, in 2018.

Before 2018, he had a career in software consulting. In 2022, he joined the Quantum Computing Competence Center, Intesa Sanpaolo, Torino, Italy, as a Quantum Software Engineer. This role allows him to pursue his passion for cutting-edge technologies.

Roberto Bertilone received the bachelor's degree in physics with the Universitá degli Studi di Messina, Messina, Italy, the master's degree in Physics of Complex Systems, from Universitá degli Studi di Torino, in 2018, and the second level master's degree in quantum computing and communication from Politecnico di Torino, Turin, Italy, in 2023.

Captivated by cutting-edge digital innovation, he gained expertise in machine learning, leading to roles in IT Governance with Intesa Sanpaolo, Torino, Italy. His research interests include designing generative AI solutions to enhance engineering productivity.

Viviana Stasino received the master's degree in condensed matter physics, in 2023, from the University of Naples "Federico II," Naples, Italy, where she is currently working toward the Ph.D. degree in quantum technologies.

During her master's thesis, she conducted research on quantum information using superconducting qubits, with a focus on both single and two-qubit gates. Her research interests include the study and the implementation of quantum algorithms in superconducting quantum processors.

Pasquale Mastrovito received the master's degree in physics from the University of Napoli "Federico II" and the Ph.D. degree in quantum technologies from the Unviersity of Napoli "Federico II", and the master's degree in mathematics from Universitá degli Studi di Torino, in 2004.

He is a Research Fellow with the UNINA Quantum Technologies Laboratory, Università degli Studi di Napoli "Federico II," Naples, Italy. He has experimentally demonstrated an on-chip tunable phase detector for superconducting qubit readout and designed novel superconducting coherent sources with fine emission control. His work also includes optimization of multiplexed qubit readout using flux-tunable travelling-wave parametric amplifiers (TWPAs), addressing intermodulation mitigation and innovative encoding of information in a dispersive readout approach. More recently, he has led projects on rapid generation of Schrödinger cat states and nondestructive detection of itinerant microwave photons, optimizing the hardware scalability. His research interests include superconducting quantum circuits, with particular emphasis on qubit control and readout, microwave photonics, and bosonic quantum computation.

Francesco Tafuri received the Ph.D. degree in physics from the University of Napoli Federico II, Naples, Italy, in 1993.

He was a visiting student with Lawrence Berkeley Laboratory, Berkeley, CA, USA, in 1990, and New York State University, Stony Brook, NY, USA, in 1992. He was a Visiting Scientist with IBM T.J. Watson Research Center, Yorktown Heights, NY, USA, in 2001. He is currently Full Professor of Condensed Matter Physics with the University of Napoli Federico II. He has authored more than 230 publications and has given more than 100 invited talks at international conferences and research centers. He is leading the Superconducting Quantum Computation Team in Napoli, which has built the first Italian Superconducting Quantum Computer. He was a member of the commission of experts on quantum technologies for the 2021–27 Italian National Research Program, and he is involved in several national and European projects involving quantum technologies. His research interests include superconductivity and quantum technologies with a special focus on weak superconductivity and superconducting electronics with significant contributions on the Josephson effect, superconducting qubits, macroscopic quantum phenomena and high critical temperature superconductors.

Davide Massarotti is currently an Associate Professor of Condensed Matter Physics with the Department of Electrical Engineering and Information Technologies, University of Naples Federico II, Naples, Italy. His research interests include the study of transport properties of unconventional Josephson devices, macroscopic quantum phenomena in superconducting circuits, and coherence and dissipation in superconducting qubits.

Leonardo Chabbra received the undergraduate degree in mechanical engineering and the master's degree in financial mathematics from the University of Warwick, Coventry, U.K., in 2008.

He is currently with G2Q Computing, Milano, Italy, working on developing compressed and innovative algorithms that can harness the power of quantum computing to solve real problems on NISQ devices. His research interests include stochastic processes, statistical mechanics, algebraic geometry, and partial differential equations (PDEs).

Davide Corbelletto is currently a Mathematician with more than twenty years of professional experience in computer science. He was a Senior Technical Advisor for several international firms, banks, and insurance companies. His role involved the management of digital transformation projects, primarily focused on the following areas: The design of reliable data architecture, the promotion of distributed and cloud computing adoption, the implementation of scalable machine learning solutions, and the broader introduction of cutting-edge products, services, and technologies. He is currently with Intesa Sanpaolo Group, Torino, Italy. In 2020, he assumed responsibility for the newly established Quantum Competence Center within the Data, AI & Technology Area of the Bank.