Loop Feynman integration on a quantum computer

Jorge J. Martínez de Lejarza[1,*] Leandro Cieri,[1,†] Michele Grossi,[2,‡] Sofia Vallecorsa,[2,§] and Germán Rodrigo[1,‖]

[1]*Instituto de Física Corpuscular, Universitat de València - Consejo Superior de Investigaciones Científicas, Parc Científic, E-46980 Paterna, Valencia, Spain*
[2]*European Organization for Nuclear Research (CERN), 1211 Geneva, Switzerland*

This work investigates in detail the performance and advantages of a new quantum Monte Carlo integrator, dubbed quantum Fourier iterative amplitude estimation (QFIAE), to numerically evaluate for the first time loop Feynman integrals in a near-term quantum computer and a quantum simulator. In order to achieve a quadratic speedup, QFIAE introduces a quantum neural network (QNN) that efficiently decomposes the multidimensional integrand into its Fourier series. For a one-loop tadpole Feynman diagram, we have successfully implemented the quantum algorithm on a real quantum computer and obtained a reasonable agreement with the analytical values. One-loop Feynman diagrams with more external legs have been analyzed in a quantum simulator. These results thoroughly illustrate how our quantum algorithm effectively estimates loop Feynman integrals and the method employed could also find applications in other fields such as finance, artificial intelligence, or other physical sciences.

I. INTRODUCTION

To unravel the mysteries of the universe at its most fundamental level, quantum field theory (QFT) stands out as an astounding theory, demonstrating significant agreement between theoretical predictions and experimental observations at particle colliders. However, as our comprehension of the particle realm progresses, the need to explore higher energies presents substantial challenges for precise measurements and theoretical predictions. In this regard, the perturbative approach in QFT has gained immense importance, emerging as a crucial framework for deriving accurate predictions in high-energy physics. Nevertheless, computing higher-order contributions in perturbative QFT is far from straightforward. The primary challenge lies in dealing with virtual quantum fluctuations, leading to the intricacies of multiloop-multileg Feynman integrals. The difficulty stems from their multidimensional nature, dependence on multiple scales, and the presence of ultraviolet (UV), infrared (IR), and threshold singularities.

Tackling these challenges often involves tiresome regularization, renormalization, and subtraction techniques.

Considering the enormous computational demands of standard methods, the imperative to seek alternative approaches becomes evident to overcome existing state-of-the-art limitations. In response, there is a growing interest in exploring innovative strategies rooted in quantum computing to address traditionally challenging problems spanning diverse domains. The potential acceleration offered by quantum computers has spurred various ideas, including Grover's algorithm for efficient database querying [1], Shor's algorithm for factorization of large integers [2], and quantum annealing for Hamiltonian minimization [3]. In the realm of particle physics, quantum algorithms have made inroads into different areas, such as lattice gauge theories [4,5]. Moreover, these algorithms have been deployed in various tasks related to high-energy colliders [6]. These applications encompass jet identification and clustering [7–10], determination [11] and integration [12] of parton densities, simulation of parton showers [13], anomaly detection [14–16], integration of elementary particle processes [17], calculation of color factors in QCD [18], and the establishment of the causal structure of multiloop Feynman diagrams [19,20]. The rapid expansion of applications underscores the versatility of quantum algorithms for a wide array of purposes [21].

In light of these recent achievements, this article seeks to investigate the potential of quantum algorithms for the efficient computation of loop Feynman integrals. Specifically, we delve into the capabilities of quantum Fourier iterative amplitude estimation (QFIAE), a quantum

[*]Contact author: jormard@ific.uv.es
[†]Contact author: leandro.cieri@ific.uv.es
[‡]Contact author: michele.grossi@cern.ch
[§]Contact author: sofia.vallecorsa@cern.ch
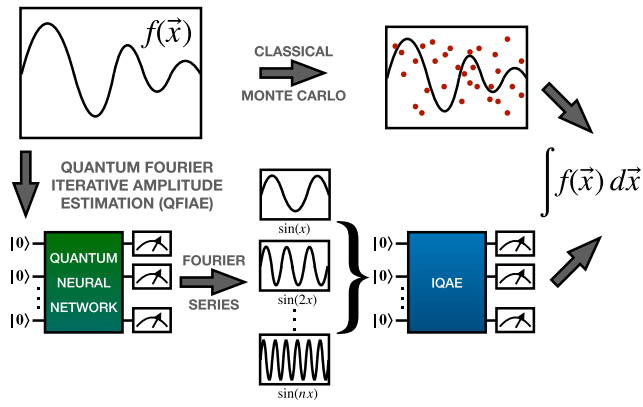[‖]Contact author: german.rodrigo@csic.es

FIG. 1. Comparison of the workflows of classical Monte Carlo integration and the QFIAE quantum algorithm.

algorithm recently introduced in [22,23]. QFIAE serves as an end-to-end quantum Monte Carlo integrator, showcasing the aimed potential for a quadratic speedup in querying the probability distribution function that follows the target function.

The workflow of QFIAE, illustrated in Fig. 1, initiates by decomposing the target function into its Fourier series using a quantum neural network (QNN) following a data reuploading approach [24–26]. As was proven in [24,27], following an exponential data encoding approach leads the quantum model to represent a truncated Fourier series. Subsequently, each trigonometric term of the Fourier series undergoes quantum integration using iterative quantum amplitude estimation (IQAE) [28], an efficient version of quantum amplitude estimation(QAE) [29].

The Fourier decomposition enables the target function to be encodable with a minimum number of quantum arithmetic operations. At the same time, it takes advantage of the fact that the sine function is more suitable for integration in a quantum approach. The QNN is the central ingredient of QFIAE that offers a viable strategy to retain the potential quadratic speedup with respect to other quantum integration algorithms recently proposed, such as Fourier quantum Monte Carlo integration (FQMCI) [30]. The key idea of FQMCI is also to use Fourier series decomposition to approximate the integrand and then estimate each component separately using QAE. Nevertheless, it relies on certain assumptions regarding the acquisition of the Fourier coefficients, which may not hold in general. When these assumptions are not met, the quantum speedup might be wiped out. The QNN ensures a reliable extraction of the coefficients in a quantum way.

The second crucial aspect of QFIAE involves leveraging the advantages of IQAE over QAE. QAE [29] is a quantum algorithm that estimates quantum state amplitudes using amplitude amplification, a generalization of Grover's algorithm [1]. This process enhances the probability of measuring a desired state over a nondesired state. However, QAE has inherent limitations, including its reliance on the

resource-intensive quantum phase estimation (QPE) subroutine [31], which involves operations considered computationally expensive for current noisy intermediate scale quantum (NISQ) devices that may compromise the anticipated quadratic speedup promised by QAE. To address this challenge, IQAE replaces QPE with a classically efficient postprocessing method, reducing the qubits and gates requirements while maintaining the asymptotic quadratic speedup.

II. QUANTUM INTEGRATION OF LOOP FEYNMAN INTEGRALS

In the realm of particle physics, loop Feynman integrals are mathematical expressions that capture quantum fluctuations arising from virtual particle interactions within subatomic physical processes. These integrals can become highly intricate, often reaching a level of complexity where analytical computation is practically impossible. Consequently, numerical integration methods involving a potentially large number of variables become necessary.

An innovative approach to address such integrals is the loop-tree duality (LTD) [32–35]. The LTD methodology transforms loops defined in the Minkowski space of loop four-momenta into trees defined in the Euclidean space of their spatial components. Additionally, it reinterprets virtual states within loops as configurations resembling real-radiation processes.

Among other advantages, this transformation provides a more intuitive understanding of the singular structure of loop integrals [36,37]. In particular, the most remarkable property of LTD is the existence of a manifestly causal representation [32,38–42], i.e., an integrand representation where certain nonphysical singularities are absent and therefore yields integrands that are numerically more stable. Since the integration domain in LTD is Euclidean, not Minkowski, it offers additional advantages, both for analytic applications such as asymptotic expansions [43,44], where the hierarchy of scales is well defined, and numerical applications [45,46] because the number of loop integration variables is independent of the number of external particles. For example, at one loop the number of independent integration variables is always three, the number of spatial components of the loop momentum, although for certain kinematic configurations this number is reduced when the dependence on any of these variables is trivial. This is the case for tadpole and bubble diagrams at one loop. Moreover, LTD offers a unified framework for cross-section calculations, since the dual representation of loop integrals in Euclidean domains allows a direct combination of virtual and real contributions at the integrand level, resulting in a fully local cancellation of IR [47–51] and UV [43,52] singularities without the need for modifying the dimensions of space-time, such as in dimensional regularization (DREG) [53,54].

We now apply QFIAE to this sort of integrals expressed in the LTD formalism, as a proof of concept of how quantum computing has the potential to handle these costly tasks in the particle physics field.

A. Tadpole loop integral

The first Feynman loop integral we address is the one-loop tadpole integral

$$\mathcal{A}_1^{(1)}(m) = \int_\ell \frac{1}{\ell^2 - m^2 + \imath 0}, \tag{1}$$

with internal mass m. The $\imath 0$ factor is the customary complex Feynman prescription for analytic continuation in different kinematical regions. The loop four-momentum to be integrated is ℓ. The corresponding mathematical expression in LTD, however, gets support in the loop three-momentum because the loop energy component is integrated out [33]. If a local UV counterterm is introduced, then its LTD representation

$$\mathcal{A}_1^{(1,R)}(m;\mu_{UV}) = -\frac{1}{2}\int_\ell \left[\frac{1}{q_{1,0}^{(+)}} - \frac{1}{q_{UV,0}^{(+)}}\left(1 + \frac{\mu_{UV}^2 - m^2}{2(q_{UV,0}^{(+)})^2}\right)\right], \tag{2}$$

is well defined in the four physical dimensions, where the integration measure reads $\int_\ell = \int d^3\ell/(2\pi)^3$. In Eq. (2), we have defined the on-shell energies $q_{1,0}^{(+)} = \sqrt{\ell^2 + m^2 - \imath 0}$ and $q_{UV,0}^{(+)} = \sqrt{\ell^2 + \mu_{UV}^2 - \imath 0}$, where μ_{UV} is the renormalization scale. The tadpole integral is one-dimensional because the integrand is independent of the solid angle, and only depends on the modulus of the three-momentum. The change of variable

$$|\boldsymbol{\ell}| = \frac{mz}{1-z}, \qquad z = [0,1), \tag{3}$$

remaps the integration variable into a finite range where the Fourier series of the target function is defined.

To implement QFIAE in a real quantum computer we must consider that quantum computers operating in the NISQ era encounter diverse sources of noise, ranging from quantum effects like decoherence to hardware-specific errors, including gate, readout, and calibration errors. At the same time, working on the same hardware technology (superconducting qubits), we advocate for a hardware-agnostic implementation strategy. Specifically, we propose an approach where the two modules of QFIAE are implemented on two different quantum computers supplied by different providers. This enables a minimization of the impact of hardware-specific noise on the overall algorithmic performance.

A one-qubit QNN is trained using an updated version of the Adam gradient descent method first presented in [55]

TABLE I. Renormalized tadpole integral $\mathcal{A}_1^{(1,R)}(m;\mu_{UV})$ on Qibo and IBMQ, as a function of the ratio of the renormalization scale μ_{UV} to the mass m, which is fixed to $m = 5$ GeV. The Fourier decomposition and IQAE integration are performed either with a quantum simulator (S) or on a real quantum device (Q). In the first raw both components are performed using classical standard methods (C).

Fourier	IQAE	$\mathcal{A}_1^{(1,R)}(m;\mu_{UV})$ $m = 5$, $\mu_{UV} = m/2$	$\mathcal{A}_1^{(1,R)}(m;\mu_{UV})$ $m = 5, \mu_{UV} = 2m$
C	C	−0.106	−0.258
S	S	−0.101(3)	−0.254(9)
S	Q	−0.108(4)	−0.270(12)
Q	S	−0.105(2)	−0.252(6)
Q	Q	−0.106(3)	−0.270(9)
Analytical		−0.1007	−0.2554

and recently improved in [56]. In this new version of the Adam algorithm, the authors propose a real-time quantum error mitigation (RTQEM) procedure, that allows to mitigate the noise in the QNN parameters during training. We use the full-stack Qibo [57] framework. The high-level algorithm has been written using Qibo, while Qibolab [58] and Qibocal [59] are used to respectively control and calibrate the 5-qubit superconducting quantum device hosted in the Quantum Research Centre (QRC) of the Technology Innovation Institute (TII).

On the other hand, a 5-qubit IQAE algorithm has been executed using Qiskit [60] on the 27-qubit IBMQ superconducting device *ibmq_mumbai*. To mitigate quantum noise during execution, we employed a pulse-efficient transpilation technique [61]. Effectively reducing the number of two-qubit gates by harnessing the hardware-native cross-resonance interaction. While it has previously demonstrated promise in the context of variational quantum circuits (VQC) [62], it has not yet been extensively explored for fault-tolerant applications. Considering the positive results presented in Table I, this work serves as an intriguing starting point for the applicability of this mitigation technique to fault-tolerant algorithms. Furthermore, we also applied two more error mitigation techniques, dynamical decoupling (DD) and zero noise extrapolation (ZNE), using the Qiskit runtime estimator primitive [63].

In particular, the hardware implementation has been done by fixing the mass to $m = 5$ GeV, and considering two values of the renormalization scale, $\mu_{UV} = 2m$ and $\mu_{UV} = m/2$. The uncertainties on the integrals presented in Table I have been computed as the quadratic sum of the individual uncertainties provided by IQAE for each trigonometric component. These uncertainties are statistical in nature and are to be combined with the statistical uncertainties from the noise of the quantum devices employed in both the QNN training and IQAE integration modules of the QFIAE algorithm. Furthermore, there are also systematic uncertainties from the classical or quantum Fourier

series truncation of the integrand function $\mathcal{A}_1^{(1,R)}(m; \mu_{UV})$. Taking into account all statistical and systematic uncertainties the obtained results are in agreement with the analytical values within uncertainties.

The results presented in Table I show a relatively small deviation from the analytical value for both $\mu_{UV} = m/2$ and $\mu_{UV} = 2m$. In particular, the agreement with the analytical values when both components of the algorithm are executed on a quantum computer (second to last row of Table I) is better than 1.7 standard deviations in all cases. This result represents a noteworthy achievement for the current state of the art in quantum computing technology. Moreover, considering that it is the first application of an end-to-end quantum algorithm executed on a quantum computer for estimating Feynman loop integrals. Furthermore, part of the deviation comes from the Fourier series approximation itself, as shown in the first row of Table I, which constitutes proof of the robustness of the quantum algorithm presented.

B. Bubble loop integral

QFIAE is also effective in dealing with Feynman loop integrals with threshold singularities. Therefore, we consider the one-loop bubble integral

$$\mathcal{A}_2^{(1)}(p, m_1, m_2) = \int_\ell \prod_{i=1}^2 \frac{1}{q_i^2 - m_i^2 + \iota 0}, \quad (4)$$

with $q_1 = \ell$ and $q_2 = \ell + p$, where p is an external momentum. As for the tadpole, we introduce a local UV counterterm to have a finite integral in the UV. The corresponding LTD representation is [32]

$$\mathcal{A}_2^{(1,R)}(p, m_1, m_2) = \int_\ell \left[\frac{1}{x_2} \left(\frac{1}{\lambda^+} + \frac{1}{\lambda^-} \right) - \frac{1}{4(q_{UV,0}^{(+)})^3} \right], \quad (5)$$

where $x_2 = \prod_{i=1,2} 2q_{i,0}^{(+)}$, $\lambda^\pm = \sum_{i=1,2} q_{i,0}^{(+)} \pm p_0$, and the on-shell energies are given by $q_{i,0}^{(+)} = \sqrt{\ell^2 + m_i^2 - \iota 0}$ with $i \in \{1, 2\}$, assuming the external momentum has vanishing
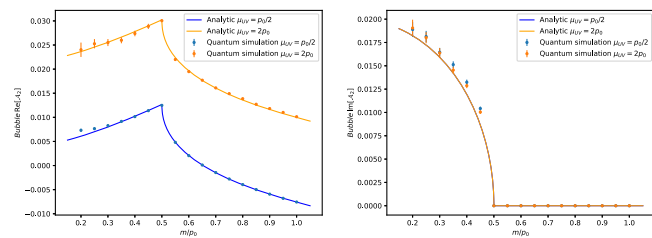

FIG. 2. Quantum integration of the real (left) and imaginary (right) part of the renormalized bubble integral $\mathcal{A}_2^{(1,R)}(p, m, m; \mu_{UV})$ as a function of the ratio of the mass m to the energy component of the external momentum set at $p_0 = 100$ GeV, and the renormalization scale μ_{UV}.

spatial components, $p = (p_0, \mathbf{0})$. If $p_0^2 < (m_1 + m_2)^2$ the integral is purely real. Otherwise, it gets an imaginary contribution from the unitary threshold singularity at $\lambda^- \to 0$, assuming $p_0 > 0$. To deal with this threshold singularity, it is convenient to introduce a contour deformation in the complex plane, in order to smooth the behavior of the function in the vicinity of the threshold without altering the result of the integral. We consider $m_1 = m_2 = m$, and proceed to estimate the integral using two different QNNs to fit separately the real and imaginary parts of the integrand and then integrate each Fourier series using IQAE. We have implemented QFIAE on two different quantum simulators. For the QNNs, we employed Pennylane [64] whereas for IQAE, we utilized Qibo.

Results are depicted in Fig. 2, where the real and imaginary parts are displayed on the left and right plots, respectively. The statistical uncertainties are calculated as in the previous section. It is worth mentioning that uncertainties in the region below the unitary threshold ($m/p_0 > 0.5$) appear to be larger compared to the rest. This is explained by the Fourier coefficients for low frequencies being larger than in the high-mass region. Hence the statistical uncertainties on the integrals of the trigonometric functions for low frequencies are intrinsically larger since the integrals we are estimating are also larger, i.e., have a larger weight, which is the Fourier coefficient. Another interesting point about these results is that in the low-mass region, the QNNs fit the target function with a slight drop in performance, hence there seems to be a correlation between the QNN struggling to fit a function and the coefficients of the lower frequency terms being larger.

Despite this, it is remarkable that the quantum integration values are in agreement with the analytical values within uncertainties. This constitutes another significant achievement since we have successfully circumvented the threshold singularity while applying a quantum algorithm to estimate the bubble integral.

C. Triangle loop integral

We now consider the one-loop three-point function, which corresponds to a triangle loop topology. The LTD representation of this integral is given by [38]

$$\mathcal{A}_3^{(1)}(p_1, p_2, m_1, m_2, m_3)$$
$$= -\int_\ell \frac{1}{x_3} \left(\frac{1}{\lambda_{12}^+ \lambda_{23}^+} + \frac{1}{\lambda_{23}^+ \lambda_{31}^-} + \frac{1}{\lambda_{31}^- \lambda_{12}^+} + (\lambda_{ij}^+ \leftrightarrow \lambda_{ij}^-) \right), \quad (6)$$

with $x_3 = \prod_{i=1}^3 2q_{i,0}^{(+)}$, where now the on-shell energies are $q_{1,0}^{(+)} = \sqrt{(\ell + \mathbf{p}_1)^2 + m_1^2 - \iota 0}$, and $q_{i,0}^{(+)} = \sqrt{\ell^2 + m_i^2 - \iota 0}$ for $i \in \{2, 3\}$. We work in the center of mass frame where $\mathbf{p}_{12} = \mathbf{p}_1 + \mathbf{p}_2 = 0$, $p_{12,0} = p_{1,0} + p_{2,0} = \sqrt{s}$ and the external momenta p_1 and p_2 are back-to-back along the z axis. The causal denominators are
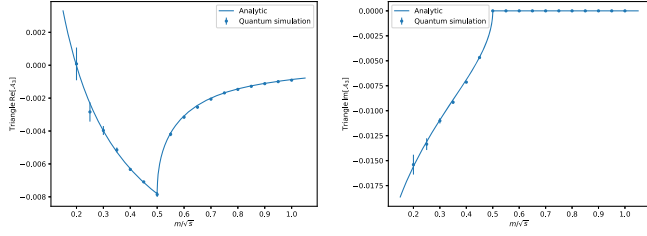
FIG. 3. Quantum integration of the real (left) and imaginary (right) part of the triangle integral $\mathcal{A}_3^{(1)}(p_1, p_2, m, m, m)$ as a function of the ratio of the mass m to the cms energy set at $\sqrt{s} = 2$ GeV.

$$\lambda_{31}^{\pm} = q_{3,0}^{(+)} + q_{1,0}^{(+)} \pm p_{1,0}, \qquad \lambda_{12}^{\pm} = q_{1,0}^{(+)} + q_{2,0}^{(+)} \pm p_{2,0},$$
$$\lambda_{23}^{\pm} = q_{2,0}^{(+)} + q_{3,0}^{(+)} \mp p_{12,0}. \tag{7}$$

The integration variables in Eq. (6) are the modulus of the loop three-momentum and its polar angle with respect to p_1, considering that the azimuthal integration is trivial. This means that the Fourier decomposition is a function of two variables and we have to integrate each of them separately. We also apply a contour deformation to deal with the unitary threshold singularity at $\lambda_{23}^{+} \to 0$, when $s > (m_2 + m_3)^2$. The contour deformation smooths the behavior of the integrand over the threshold singularity and therefore significantly improves the quality of the Fourier decomposition. The estimations of the triangle integral, obtained in Pennylane and Qibo simulators, are shown in Figs. 3(left) and 3(right), illustrating the real and imaginary components, respectively. The statistical uncertainties are calculated as in the previous sections and are expected to be higher since we are performing a double IQAE integral, and each integration introduces an error. However, the estimated uncertainties align well with the deviations observed in the real and imaginary components of the integral. All in all, this represents another noteworthy accomplishment as we have successfully extended for the first time the QFIAE algorithm to a two-dimensional function with a threshold singularity, and utilized a quantum algorithm to approximate the integral with sufficient accuracy.

III. CONCLUSIONS

In this work, we have for the first time successfully implemented a quantum algorithm for evaluating Feynman loop integrals in quantum hardware. Our approach is based on the nonconventional LTD representation of Feynman loop integrals and the QFIAE quantum algorithm which provides a theoretical quadratic quantum speedup in the number of queries of the probability distribution function.

We have provided a clear indication of how the optimization of a quantum circuit, at the level of pulses and the application of combined error mitigation techniques, enable the collection of accurate results. Specifically, we employed QFIAE to numerically integrate the one-dimensional tadpole Feynman loop integral entirely on real quantum hardware provided by Qibo and IBM. Moreover, loop Feynman integrals with more external legs and dimensions have also been integrated using QFIAE on quantum simulators across various kinematical regions, circumventing different threshold singularities using a contour deformation. The obtained results demonstrate the remarkable performance of QFIAE in estimating these integrals.

Overall, our study highlights the potential of quantum Monte Carlo integrators, particularly the QFIAE quantum algorithm, in integrating Feynman loop integrals. These findings, some of them obtained from quantum superconducting devices, serve as an initial step toward the search for a quantum method capable of speeding up the computationally expensive task of evaluating Feynman multiloop-multileg integrals. Moreover, they offer a proof of concept for the method's applicability to wider challenges across various domains that require Monte Carlo integration, such as finance, artificial intelligence and other physical sciences.

When considering the scalability of QFIAE to higher dimensions, it is important to analyze its two key components separately. Regarding the IQAE integration, the use of Fourier decomposition allows us to express a multidimensional integral as a product of individual integrals containing Fourier terms with independent integration variables. Consequently, regardless of the dimensionality, IQAE is always applied to one-dimensional trigonometric integrands after Fourier decomposition. This ensures its efficient extension to multidimensional functions. In relation to the QNN fitting the target function, universality has been theoretically demonstrated in [24] for exponential encoding and a data re-uploading scheme. This constitutes a theoretical proof that the first component of QFIAE is also scalable to higher dimensions.

Still, the main challenge lies in identifying a suitable ansatz capable of ensuring the trainability of multidimensional functions, which is essential for addressing higher-loop integrals. While universality has proven effective in lower dimensions, extending it to higher dimensions remains an open problem. In this work, we achieved a successful fit for one and two-dimensional functions. However, the search for a viable ansatz in higher dimensions which involves a tradeoff between trainability and expressivity of the variational quantum circuit is left for future work.

Progressing in the previous limitation would assure, according to the algorithm proposed in this article, a significant noteworthy advancement over classical Monte Carlo methods for computing these integrals, presenting new opportunities for unlocking previously inaccessible levels of precision in theoretical predictions for high-energy physics at colliders.

ACKNOWLEDGMENTS

$$\mathcal{L}_{LA}^{(l)}(\vec{x},\vec{\theta}) = \prod_{i=1}^{M} R_z(\theta_3 x_i + \kappa\theta_4)R_y(\theta_1 x_i + \theta_2), \quad \text{with} \quad \begin{cases} \kappa = 1 & \text{if } l \text{ is the last layer,} \\ \kappa = 0 & \text{otherwise.} \end{cases} \tag{A1}$$

On the other hand, the IQAE has been executed using Qiskit [60] on the IBMQ 27-qubits device *ibmq_mumbai*. The quantum circuits of the operators $\mathcal{A}$ and $\mathcal{Q}$ of the IQAE algorithm are presented in Fig. 4. To mitigate quantum noise during the execution of this algorithm, we employed a pulse-efficient transpilation technique [61]. This technique effectively reduces the number of two-qubit gate operations by harnessing the hardware-native cross-resonance interaction, potentially leading to a reduction in quantum noise. While it has previously demonstrated promise in the context of Variational Quantum Circuits (VQC) [62], it has not yet been extensively explored for fault-tolerant applications. Hence, this work serves as an intriguing starting point for the applicability of this mitigation technique to fault-tolerant algorithms, given the positive results presented in Fig. 3. Furthermore, we also applied two more error mitigation techniques, Dynamical Decoupling (DD) and Zero Noise Extrapolation (ZNE), which are automatized within the Qiskit Runtime Estimator primitive [63].

APPENDIX B: SIMULATION IMPLEMENTATION OF QFIAE

We have also implemented the QFIAE algorithm using two different simulation frameworks. Pennylane [64] has

APPENDIX A: HARDWARE IMPLEMENTATION OF QFIAE

Although the implementation of Monte Carlo integration in quantum simulators is of great interest for proof of concept purposes, the potential quantum advantage will only be materialized when the quantum algorithm is run on a real quantum device. To this aim, we have addressed the challenge of implementing an end-to-end quantum integrator method into two different quantum devices.

First, the QNN has been trained using an updated version of the Adam gradient descent method first presented in [55] and recently improved in [56]. In this new version of the algorithm, the authors propose a Real-Time Quantum Error Mitigation (RTQEM) procedure, that allows to mitigate the noise in the QNN parameters during training. We use the Qibo [57] framework to compute the full-stack procedure. The high-level quantum computing algorithm has been written using Qibo, while Qibolab [58] and Qibocal [59] are used to respectively control and calibrate the 5-qubit superconducting device employed.

In this case, a more hardware-friendly linear ansatz has been chosen to construct the QNN in one qubit. In particular each layer $\mathcal{L}_{LA}^{(l)}(\vec{x},\vec{\theta})$ is defined as:

been used for QNN implementation, whereas Qibo [57] has been used for applying IQAE to the Fourier series.

In particular, the linear ansatz corresponding to each layer $\mathcal{L}_{LA}^{(l)}(\vec{x},\vec{\theta})$ for training the QNN to fit a M-dimensional function is the following [25]:

$$\mathcal{L}_{LA}^{(l)}(\vec{x},\vec{\theta}) = \prod_{i=1}^{M} \mathcal{S}(x_i)\mathcal{A}_i^{(l)}(\vec{\theta}_{l,i}), \tag{B1}$$

where $\mathcal{S}(x_i)$ and $\mathcal{A}_i^{(l)}$ are chosen as:

$$\mathcal{S}(x_i) = R_z(x_i), \quad \mathcal{A}_i^{(l)}(\vec{\theta}_{l,i}) = R_z(\theta_{l,i,1})R_y(\theta_{l,i,2})R_z(\theta_{l,i,3}). \tag{B2}$$

Once the Fourier coefficients are obtained from the QNN, we implement the IQAE algorithm. To uphold the claimed quantum advantage provided by Grover's amplitude amplification, certain conditions must be fulfilled. First, the probability distribution of the functions to be integrated should be encodable into a shallow quantum circuit. In view of this requirement, we will use the distribution $p(x_i) = 1/2^n$ generated by applying an n-dimensional Hadamard gate, denoted as $\mathcal{H}^{\otimes n}$, which corresponds to a quantum circuit of depth 1. The second
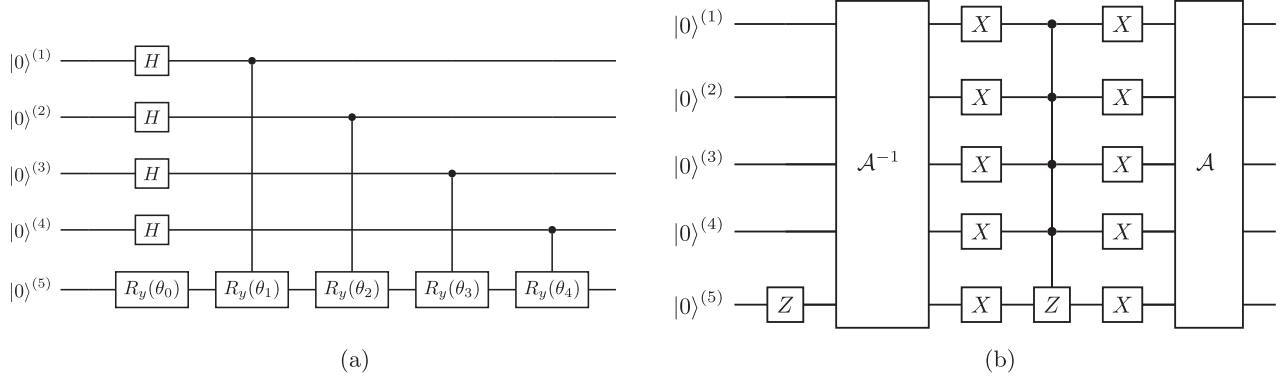
FIG. 4. Quantum circuits of the operators for the IQAE component of the QFIAE algorithm. (a) Amplitude operator $\mathcal{A}$. (b) Amplification operator $\mathcal{Q}$.

condition is that the target function has to be encodable with a minimum number of quantum arithmetic operations. That will be achieved selecting the target function as a $\sin(x_i)^2$ to be integrated in $[x_{i,\min}, x_{i,\max}]$. Then the integrals of the Fourier terms are obtained from the integral of the sine function.

Under these considerations and choosing $n_{\text{qubits}} = 5$, the quantum circuits corresponding to the $\mathcal{A}$ and $\mathcal{Q}$ operators are shown in Fig. 4. Note that in Fig. 4(a) the rotation angles encode the information about the limits of integration $x_{i,\min}$ and $x_{i,\max}$. In particular, they are defined as:

$$\theta_0 = (x_{\max} - x_{\min})/2^n + 2x_{\min}, \quad \theta_i = 2^{(i+1)}(x_{\max} - x_{\min})/2^n,$$
$$n = n_{\text{qubits}}. \tag{B3}$$

For more information, a tutorial on QFIAE implementation can be found at [23].

APPENDIX C: PARAMETERS EMPLOYED FOR INTEGRALS IMPLEMENTATION

See Tables II and III below:

TABLE II. Parameters for training the QNN.

	Tadpole (HW)	Tadpole (SIM)	Bubble $m/p_0 < 0.5$ (SIM)	Bubble $m/p_0 > 0.5$ (SIM)	Triangle $m/\sqrt{s} < 0.5$ (SIM)	Triangle $m/\sqrt{s} > 0.5$ (SIM)
Layers	3	5	10	20	10	10
n_Fourier	5	5	10	20	10[a]	10
Step_size	0.100	0.060	0.095	0.020	0.065	0.045
Max_steps	60	300	450	200	450	400
Data_train	15	150	150	1500	3600	10000
Shots	500	⋯	⋯	⋯	⋯	⋯
δ	⋯	⋯	21	⋯	0.1	⋯

[a]Note that since it is a 2D function, the total number of Fourier coefficients is $nFourier^2 = 100$.

TABLE III. Parameters for the IQAE.

	Tadpole (HW & SIM)	Bubble (SIM)	Triangle (SIM)
ε	0.01	0.001	0.001
α	0.05	0.05	0.05
Shots	100000	1000	10000

[1] L. K. Grover, Phys. Rev. Lett. **79**, 325 (1997).

[2] P. W. Shor, SIAM J. Comput. **26**, 1484 (1997).

[3] T. Kadowaki and H. Nishimori, Phys. Rev. E **58**, 5355 (1998).

[4] E. Zohar, J. I. Cirac, and B. Reznik, Rep. Prog. Phys. **79**, 014401 (2015).

[5] S. P. Jordan, K. S. M. Lee, and J. Preskill, Science **336**, 1130 (2012).

[6] A. Delgado *et al.*, in *Snowmass 2021* (2022), arXiv:2203.08805.

[7] A. Y. Wei, P. Naik, A. W. Harrow, and J. Thaler, Phys. Rev. D **101**, 094015 (2020).

[8] A. Delgado and J. Thaler, Phys. Rev. D **106**, 094016 (2022).

[9] J. J. Martínez de Lejarza, L. Cieri, and G. Rodrigo, Phys. Rev. D **106**, 036021 (2022).

[10] J. J. M. de Lejarza, L. Cieri, and G. Rodrigo, Proc. Sci. ICHEP2022 (**2022**) 241 [arXiv:2209.08914].

[11] A. Pérez-Salinas, J. Cruz-Martinez, A. A. Alhajri, and S. Carrazza, Phys. Rev. D **103**, 034027 (2021).

[12] J. M. Cruz-Martinez, M. Robbiati, and S. Carrazza, Quantum Sci. Technol. **9**, 035053 (2024).

[13] K. Bepari, S. Malik, M. Spannowsky, and S. Williams, Phys. Rev. D **106**, 056002 (2022).

[14] K. A. Woźniak, V. Belis, E. Puljak, P. Barkoutsos, G. Dissertori, M. Grossi, M. Pierini, F. Reiter, I. Tavernelli, and S. Vallecorsa, arXiv:2301.10780.

[15] J. Schuhmacher, L. Boggia, V. Belis, E. Puljak, M. Grossi, M. Pierini, S. Vallecorsa, F. Tacchino, P. Barkoutsos, and I. Tavernelli, Mach. Learn. **4**, 045031 (2023).

[16] E. Bermot, C. Zoufal, M. Grossi, J. Schuhmacher, F. Tacchino, S. Vallecorsa, and I. Tavernelli, in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)* (2023), Vol. 1, pp. 331–341, 10.1109/QCE57702.2023.00045.

[17] G. Agliardi, M. Grossi, M. Pellen, and E. Prati, Phys. Lett. B **832**, 137228 (2022).

[18] H. A. Chawdhry and M. Pellen, SciPost Phys. **15**, 205 (2023).

[19] S. Ramírez-Uribe, A. E. Rentería-Olivo, G. Rodrigo, G. F. R. Sborlini, and L. Vale Silva, J. High Energy Phys. 05 (2022) 100.

[20] G. Clemente, A. Crippa, K. Jansen, S. Ramírez-Uribe, A. E. Rentería-Olivo, G. Rodrigo, G. F. R. Sborlini, and L. Vale Silva, Phys. Rev. D **108**, 096035 (2023).

[21] A. D. Meglio *et al.*, PRX Quantum **5**, 037001 (2024).

[22] J. J. M. de Lejarza, M. Grossi, L. Cieri, and G. Rodrigo, in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)* (2023), Vol. 1 pp. 571–579, 10.1109/QCE57702.2023.00071.

[23] J. J. M. de Lejarza, Tutorial: Quantum Fourier Iterative Amplitude Estimation, https://qibo.science/qibo/stable/code-examples/tutorials/qfiae/qfiae_demo.html (2023).

[24] M. Schuld, R. Sweke, and J. J. Meyer, Phys. Rev. A **103**, 032430 (2021).

[25] B. Casas and A. Cervera-Lierta, Phys. Rev. A **107**, 062612 (2023).

[26] A. Pérez-Salinas, A. Cervera-Lierta, E. Gil-Fuster, and J. I. Latorre, Quantum **4**, 226 (2020).

[27] F. J. Gil Vidal and D. O. Theis, Front. Phys. **8**, 297 (2020).

[28] D. Grinko, J. Gacon, C. Zoufal, and S. Woerner, npj Quantum Inf. **7**, 52 (2021).

[29] G. Brassard, P. Høyer, M. Mosca, and A. Tapp, Quantum amplitude amplification and estimation (2002), 10.1090/conm/305/05215.

[30] S. Herbert, Quantum **6**, 823 (2022).

[31] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University Press, Cambridge, England, 2011).

[32] J. J. "Aguilera-Verdugo, F. Driencourt-Mangin, R. J. Hernández-Pinto, J. Plenter, S. Ramirez-Uribe, A. E. Renteria Olivo, G. Rodrigo, G. F. R. Sborlini, W. J. Torres Bobadilla, and S. Tracz, Phys. Rev. Lett. **124**, 211602 (2020).

[33] S. Catani, T. Gleisberg, F. Krauss, G. Rodrigo, and J.-C. Winter, J. High Energy Phys. 09 (2008) 065.

[34] I. Bierenbaum, S. Catani, P. Draggiotis, and G. Rodrigo, J. High Energy Phys. 10 (2010) 073.

[35] I. Bierenbaum, S. Buchta, P. Draggiotis, I. Malamos, and G. Rodrigo, J. High Energy Phys. 03 (2013) 025.

[36] S. Buchta, G. Chachamis, P. Draggiotis, I. Malamos, and G. Rodrigo, J. High Energy Phys. 11 (2014) 014.

[37] J. J. Aguilera-Verdugo, F. Driencourt-Mangin, J. Plenter, S. Ramírez-Uribe, G. Rodrigo, G. F. R. Sborlini, W. J. Torres Bobadilla, and S. Tracz, J. High Energy Phys. 12 (2019) 163.

[38] J. J. Aguilera-Verdugo, R. J. Hernandez-Pinto, G. Rodrigo, G. F. R. Sborlini, and W. J. Torres Bobadilla, J. High Energy Phys. 01 (2021) 069.

[39] S. Ramírez-Uribe, R. J. Hernández-Pinto, G. Rodrigo, G. F. R. Sborlini, and W. J. Torres Bobadilla, J. High Energy Phys. 04 (2021) 129.

[40] J. Jesús Aguilera-Verdugo, R. J. Hernández-Pinto, G. Rodrigo, G. F. R. Sborlini, and W. J. Torres Bobadilla, J. High Energy Phys. 02 (2021) 112.

[41] G. F. R. Sborlini, Phys. Rev. D **104**, 036014 (2021).

[42] W. J. Torres Bobadilla, J. High Energy Phys. 04 (2021) 183.

[43] F. Driencourt-Mangin, G. Rodrigo, and G. F. R. Sborlini, Eur. Phys. J. C **78**, 231 (2018).

[44] J. Plenter and G. Rodrigo, Eur. Phys. J. C **81**, 320 (2021).

[45] S. Buchta, G. Chachamis, P. Draggiotis, and G. Rodrigo, Eur. Phys. J. C **77**, 274 (2017).

[46] F. Driencourt-Mangin, G. Rodrigo, G. F. R. Sborlini, and W. J. Torres Bobadilla, Phys. Rev. D **105**, 016012 (2022).

[47] R. J. Hernandez-Pinto, G. F. R. Sborlini, and G. Rodrigo, J. High Energy Phys. 02 (2016) 044.

[48] G. F. R. Sborlini, F. Driencourt-Mangin, R. Hernandez-Pinto, and G. Rodrigo, J. High Energy Phys. 08 (2016) 160.

[49] G. F. R. Sborlini, F. Driencourt-Mangin, and G. Rodrigo, J. High Energy Phys. 10 (2016) 162.

[50] R. M. Prisco and F. Tramontano, J. High Energy Phys. 06 (2021) 089.

[51] J. de Jesús Aguilera-Verdugo *et al.*, Symmetry **13**, 1029 (2021).

[52] F. Driencourt-Mangin, G. Rodrigo, G. F. R. Sborlini, and W. J. Torres Bobadilla, J. High Energy Phys. 02 (2019) 143.

[53] C. G. Bollini and J. J. Giambiagi, Nuovo Cimento B **12**, 20 (1972).

[54] G. 't Hooft and M. J. G. Veltman, Nucl. Phys. **B44**, 189 (1972).

[55] M. Robbiati, S. Efthymiou, A. Pasquale, and S. Carrazza, Proc. Sci., ICHEP2022 (**2022**) 206 [arXiv:2210.10787].

[56] M. Robbiati, A. Sopena, A. Papaluca, and S. Carrazza, arXiv:2311.05680.

[57] S. Efthymiou, S. Ramos-Calderer, C. Bravo-Prieto, A. Pérez-Salinas, D. García-Martín, A. Garcia-Saez, J. I. Latorre, and S. Carrazza, Quantum Sci. Technol. **7,** 015018 (2021).

[58] S. Efthymiou *et al.*, Quantum **8,** 1247 (2024).

[59] A. Pasquale, S. Efthymiou, S. Ramos-Calderer, J. Wilkens, I. Roth, and S. Carrazza, in *21th International Workshop on Advanced Computing and Analysis Techniques in Physics Research: AI meets Reality* (2023), arXiv:2303.10397.

[60] Qiskit contributors, Qiskit: An open-source framework for quantum computing (2023), 10.5281/zenodo.2573505.

[61] N. Earnest, C. Tornow, and D. J. Egger, Phys. Rev. Res. **3,** 043088 (2021).

[62] A. Melo, N. Earnest-Noble, and F. Tacchino, Quantum **7,** 1130 (2023).

[63] IBMQ, Qiskit runtime estimator primitive service, https://qiskit.org/ecosystem/ibm-runtime/stubs/qiskit_ibm_runtime.Estimator.html (2023) (accessed: 2023-11-3).

[64] V. Bergholm *et al.*, arXiv:1811.04968.