Efficient digitized counterdiabatic quantum optimization algorithm within the impulse regime for portfolio optimization

Alejandro Gomez Cadavid,[1,2] Iraitz Montalban,[1,3] Archismita Dalal,[1] Enrique Solano,[1,*] and Narendra N. Hegade[1,†]

[1] *Kipu Quantum, Greifswalderstrasse 212, 10405 Berlin, Germany*

[2] *Department of Physical Chemistry, University of the Basque Country UPV/EHU, Apartado 644, 48080 Bilbao, Spain*

[3] *Department of Physics, University of the Basque Country UPV/EHU, Barrio Sarriena, s/n, 48940 Leioa, Biscay, Spain*

(Received 10 October 2023; revised 19 August 2024; accepted 3 October 2024; published 13 November 2024)

We propose a faster digital quantum algorithm for portfolio optimization using the digitized counterdiabatic quantum optimization paradigm in the impulse regime, that is, where the counterdiabatic terms are dominant. Our approach notably reduces the circuit depth requirement of the algorithm and enhances the solution accuracy, making it suitable for current quantum processors. We apply this protocol to a real-case scenario of portfolio optimization with 20 assets, using purely quantum and hybrid classical-quantum paradigms. We demonstrate the advantages of our protocol using up to 20 qubits on an IonQ trapped-ion quantum computer. By benchmarking our method against the standard quantum approximate optimization algorithm and finite-time digitized adiabatic algorithms, we obtain a significant reduction in the circuit depth by factors of 2.5 to 40, while minimizing the dependence on the classical optimization subroutine. Besides portfolio optimization, the proposed method is applicable to a large class of combinatorial optimization problems.

DOI: 10.1103/PhysRevApplied.22.054037

I. INTRODUCTION

Quantum computing aim to challenge the status quo of classical computation for hard-to-solve problems, as was first demonstrated in Shor's quantum algorithm for prime factorization [1]. However, the hardware requirements raise valid concerns over scalability issues and the effective translation from theoretical to experimental realizations [2]. In this sense, when using noisy intermediate-scale quantum (NISQ) processors, it is unavoidable to consider specific-purpose implementations of the targeted problem [3]. Our NISQ era needs pragmatic solutions via the co-designed encoding of compressed algorithms in available hardware.

Combinatorial optimization problems relate to many challenges faced by companies in various industries, such as logistics, finance, and chemistry [4,5]. This explains the wide attention and methods developed, where the quantum approximate optimization algorithm (QAOA) [6] is one of the most adopted variational protocols [7]. The QAOA is one of the hybrid classical-quantum approaches known as variational quantum algorithms, and it requires classical training to fit an ansatz or parametrized quantum circuit (PQC) composed of discrete gates. Similarly to Shor's algorithm, it suffers from discrepancies between theoretical proposals [8,9] and hardware implementations [10]. Given its canonical form, it produces relatively deep quantum circuits that are known to be impacted by hardware noise [11]. There is an extensive literature on how PQCs can be trained and modified in order to circumvent those issues [12–14].

The prospect of achieving a quantum advantage in finance use cases is being extensively explored [4,15]. In particular, the portfolio optimization problem is a non-trivial and interesting use case for NISQ testbeds with different strategies [16–18]. Both analog (quantum annealing) and digital (QAOA) algorithms are commonly used for solving this constrained optimization problem. The state-of-the-art implementation of QAOA for portfolio optimization shows the convenience of using a constraint-preserving mixer Hamiltonian, which strongly improves the approximation ratio [18].

In this paper we present an efficient quantum computing solution for portfolio optimization using a fast, purely quantum, digitized counterdiabatic quantum optimization (DCQO) protocol, as well as its hybrid counterpart (h-DCQO). Our results show an important improvement on

*Contact author: enrique.solano@kipu-quantum.com
†Contact author: narendra.hegade@kipu-quantum.com

 © 2024 American Physical Society

previous results [19–24], as we will explain in detail. We rely on the concept of the impulse regime, which corresponds to the fast nonadiabatic evolution where the counterdiabatic (CD) terms are dominant. Our method reduces the circuit depth and increases the solution accuracy with respect to finite-time digitized adiabatic algorithms and the conventional QAOA. In particular, we successfully test our protocol for a 20-asset portfolio optimization problem on IonQ quantum processors.

The paper is organized as follows. In Sec. II we briefly describe the problem of portfolio optimization, the techniques of adiabatic quantum optimization, CD protocols, and their digitized versions. We also discuss QAOA and its extension using CD protocols. In Sec. III we elaborate on our proposed DCQO and h-DCQO algorithms and introduce our modified definition of the approximation ratio. Then we present our results, from both simulations and cloud-based demonstrations, and discuss them in Sec. IV. Finally, in Sec. V we present our conclusions.

II. PORTFOLIO OPTIMIZATION AND DCQO

A. Portfolio optimization

Within the financial services and banking industry that emerged with Markowitz's seminal paper in 1952 [25], portfolio optimization occupies a central role. It describes a set of n assets $x_i \in \{X\}_n$ from which a subset needs to be picked. The goal is to maximize the revenue while minimizing the risk at t future timesteps. Each asset i has an associated revenue forecast at a given time period (e_i) and the covariance between assets (c_{ij}) sets the risk amount in terms of diversification.

For the sake of simplicity, we focus on a single-timestep version of this problem with Boolean asset investment. Let us consider, then, a single timestep t and a budget B, which defines the number of assets that can be selected. The budget sets a constraint on the problem, which is described by the summation cost x_i of investing on each individual asset i. Ideally, the solution should not exceed this budget condition ($\sum_i x_i \leq B$). Therefore, the problem is posed in the first stage as follows:

$$\begin{aligned} \max_{x_i\in\{0,1\}} \quad & \sum_{i=1}^{n} x_i e_i - \theta \sum_{i,j=1}^{n} x_i x_j c_{ij} \\ \text{s.t.} \quad & \sum_{i=1}^{n} x_i \leq B\,. \end{aligned} \tag{1}$$

Here, $x_i \in \{0,1\}$ is the mask associated with the selection of our set of assets, while θ is a Lagrange multiplier modulating the amount of risk we would like to assume in our investments.

Single-period discrete portfolio optimization has been shown to be NP-complete regardless of the risk [26]. Several classical solutions have been proposed for this problem. However, quantum computing may produce faster solutions due to the connection between spin glasses and Markowitz portfolio optimization [15].

To convert our canonical formulation of Eq. (1) to a Hamiltonian formulation, the problem is reformulated as follows:

$$\max_{x_i\in\{0,1\}} \quad \theta_1 \sum_{i=1}^{n} x_i e_i \;-\; \theta_2 \sum_{i,j=1}^{n} x_i x_j c_{ij} \;-\; \theta_3 \left(\sum_{i=1}^{n} x_i - B \right)^2 . \tag{2}$$

Here, the budget constraint has been integrated into the cost function and $\theta_{1,2,3}$ are Lagrange multipliers, which determine the relevance of each term. In this way one can modulate not only the risk but also the option of exceeding the budget if necessary, being able to simulate all potential scenarios for a decision to be made regarding the portfolio. The objective function in Eq. (2) can then easily be translated into an Ising formulation that simplifies the encoding [27,28] of the problem in future steps.

B. Adiabatic quantum optimization and CD protocols

Computing the optimal solution for Eq. (2) requires finding the ground state of an Ising spin-glass Hamiltonian H_f, which we call the problem Hamiltonian:

$$H_f = \sum_{i=1}^{n} h_i Z_i + \sum_{i<j} J_{ij} Z_i Z_j\,. \tag{3}$$

The explicit form of the coefficients h_i and J_{ij} can be found in Appendix A. The ground state of H_f encodes the binary string, or strings in the case of degeneracies, as the mask of assets to be chosen. In this sense, we use the adiabatic theorem to find this ground state [29,30]. Here, a quantum system slowly evolves from the ground state of an initial Hamiltonian, chosen in this case as $H_i = -\sum_{i=1}^{n} X_i$ with ground state $\left((|0\rangle + |1\rangle)/\sqrt{2}\right)^{\otimes n}$, towards the ground state of the problem Hamiltonian H_f. This evolution is governed by the Hamiltonian

$$H_{\text{ad}}(t) = [1 - \lambda(t)]\, H_i + \lambda(t) H_f\,, \tag{4}$$

where $\lambda(t)$ is a scheduling function satisfying the conditions $\lambda(0) = 0$ and $\lambda(T) = 1$, with T the evolution time. In this manner, it is ensured that the system goes from the ground state of H_i to the ground state of H_f. As a consequence, the measured state at the end of the process will yield the solution to the optimization problem. In this paper we use the scheduling function $\lambda(t) = \sin^2\left[\pi/2 \sin^2(\pi t/2T)\right]$, such that $\dot{\lambda}(0) = \dot{\lambda}(T) = 0$.

The process described in Eq. (4) can be encoded in quantum annealers, which are analog devices that have been extensively used in order to tackle various optimization problems [31]. By nature, quantum annealing is a slow process, therefore affected by decoherence, which leads to an accumulation of errors. To mitigate this issue, one might resort to rapid nonadiabatic evolution. However, performing fast evolution of Eq. (4) causes excitations between eigenstates, compromising the quality of the final solution.

Counterdiabatic driving is a method that helps to suppress nonadiabatic transitions due to rapid evolution [32–34]. This driving is introduced as an extension of Eq. (4) and can be written in the form

$$H(t) = H_{\text{ad}}(t) + \dot{\lambda}(t) A_{\lambda}. \tag{5}$$

Here, A_{λ} is known as the adiabatic gauge potential, responsible for suppressing the nonadiabatic excitations. The calculation of the gauge potential is computationally expensive since it requires knowledge of the instantaneous eigenspectrum of H_{ad} [33]. However, taking the proposal from Refs. [35–38] one can construct approximate CD protocols that can be easily obtained and experimentally realized. For instance, the gauge potential from Eq. (5) can be approximated by a series of nested commutators (NC),

$$A_{\lambda}^{(l)} = i \sum_{k=1}^{l} \alpha_k(t) \underbrace{[H_{\text{ad}}, [H_{\text{ad}}, \ldots [H_{\text{ad}}}_{2k-1}, \partial_{\lambda} H_{\text{ad}}]]], \tag{6}$$

where $\alpha(t)$ is a CD coefficient to be found and l sets the order of approximation of the gauge potential. For example, for the Hamiltonians H_i and H_f mentioned before, the first-order ($l = 1$) approximation of the gauge potential is

$$A_{\lambda}^{(1)} = -2\alpha_1 \left[\sum_i h_i Y_i + \sum_{i<j} J_{ij} \left(Y_i Z_j + Z_i Y_j \right) \right],$$

where

$$\alpha_1 = -\frac{1}{4} \frac{\sum_i h_i^2 + \sum_{i<j} J_{ij}^2}{(1-\lambda)^2 \left(\sum_i h_i^2 + 4 \sum_{i \neq j} J_{ij}^2 \right) + \lambda^2 \left[\sum_i h_i^4 + \sum_{i \neq j} J_{ij}^4 + 6 \sum_{i \neq j} h_i^2 J_{ij}^2 + 6 \sum_{i<j<k} \left(J_{ij}^2 J_{ik}^2 + J_{ij}^2 J_{jk}^2 + J_{ik}^2 J_{jk}^2 \right) \right]}. \tag{7}$$

The detailed calculation of the analytical expression for α_1 is given in Appendix B. Higher orders can be computed similarly. However, as the order in Eq. (6) increases, the locality of the operators also increases, making experimental implementation less feasible. For example, the second-order approximation contains three-body and four-body operators, which would need to be further decomposed into two-body operators, resulting in a larger circuit depth and increased gate counts.

To efficiently implement the counterdiabatic protocols from Eqs. (5) and (6), we adopt a digital approach, discretizing the total evolution time T into N small steps, each of duration dt. During each brief time interval, the Hamiltonian can be viewed as time-independent. This method allows for the realization of arbitrary counterdiabatic terms, which are not achievable with current analog quantum computers. By employing first-order trotterization, the continuous-time dynamics can be approximated as

$$U_{\text{digital}} = \prod_{m=1}^{N} \prod_{k=1}^{L} \exp\left[-i\, dt\, c_k(mdt)\, H_k \right]. \tag{8}$$

Here, H_k stands for each k-local Pauli operator corresponding to the complete Hamiltonian expressed as $H(t) = \sum_{k=1}^{L} c_k(t) H_k$, with L being the number of terms of $H(t)$. Each matrix exponential term in the above expression can be easily decomposed into a set of quantum gates. The accuracy of this digitization depends on the step size dt, with smaller values generally offering improved fidelity at the cost of requiring a larger number of gate sequences. The accuracy and robustness of this digitization method, for both digitized adiabatic evolution and digitized counterdiabatic evolution, have been examined in recent studies [39]. Furthermore, more efficient methods known as commutator product formulas have recently been introduced to realize digitized counterdiabatic evolution [40].

C. Hybrid approaches

Purely quantum approaches may often require long circuit depths resulting from long evolution times, thus exceeding the coherence time of the qubits. This challenge led to the development of hybrid classical-quantum approaches, harnessing the power of classical and quantum devices while looking for optimal solutions using shorter quantum circuits. This will greatly reduce the cumulative effect of noise on existing hardware.

In Ref. [6], the authors proposed the QAOA algorithm, which is inspired by an adiabatic quantum computing procedure that approximates the solution to a combinatorial problem by a set of discrete steps. This is done by combining a problem Hamiltonian and a mixer Hamiltonian, respectively equivalent to H_f and H_i in Eq. (4). This algorithm has in recent years become the standard technique for most combinatorial problems tackled on near-term quantum devices, besides quantum annealers [31].

The QAOA algorithm starts by initializing the qubits in the ground state of H_i, which is a superposition of all possible 2^n states, $|+\rangle^{\otimes n}$. Then, following alternating layers of the unitary associated to the problem Hamiltonian $U_f(\gamma) = e^{-i\gamma H_f}$, parameterized by γ, and a mixer unitary $U_i(\beta) = e^{-i\beta H_i}$, parameterized by β, the algorithm produces the following state:

$$|\boldsymbol{\gamma}, \boldsymbol{\beta}\rangle = U_i(\beta_p)U_f(\gamma_p)\ldots U_i(\beta_1)U_f(\gamma_1)|+\rangle^{\otimes n}. \quad (9)$$

In the same fashion, the hybrid counterpart of the evolution associated with Eq. (5) introduces a new block to the alternating layers that account for the counterdiabatic term $U_c(\alpha)$, which is then parametrized by α,

$$|\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}\rangle = U_c(\alpha_p)U_i(\beta_p)U_f(\gamma_p)\ldots U_c(\alpha_1)U_i(\beta_1)U_f(\gamma_1)|+\rangle^{\otimes n}.$$

The extension of QAOA using CD protocols, known as DC-QAOA, was introduced in Refs. [22,41,42], and has been applied to various problems, such as molecular docking [24]. Recent research also highlights its improved performance when using higher-order CD terms [23]. In the next section, we will explain how the DC-QAOA method can be further refined by building the variational ansatz using only the CD term. This approach significantly reduces the circuit depth, resulting in better performance, particularly when run on real hardware.

III. METHODOLOGY

In this section we elaborate on the design and analysis of counterdiabatic protocols in the impulse regime for the problem of portfolio optimization. We will describe the two implementations of digitized counterdiabatic quantum optimization, one being purely quantum and the other being a quantum-classical hybrid algorithm. Finally, we explain how we model the portfolio optimization problem and propose an efficient quantum algorithm.

A. DCQO in the impulse regime

We analyze the evolution of the CD-assisted Hamiltonian (5) in three regimes, namely impulse, intermediate, and adiabatic regimes [45,46]. These regimes are characterized by the total evolution time, with short (long) times falling in the impulse (adiabatic) regime. In the impulse

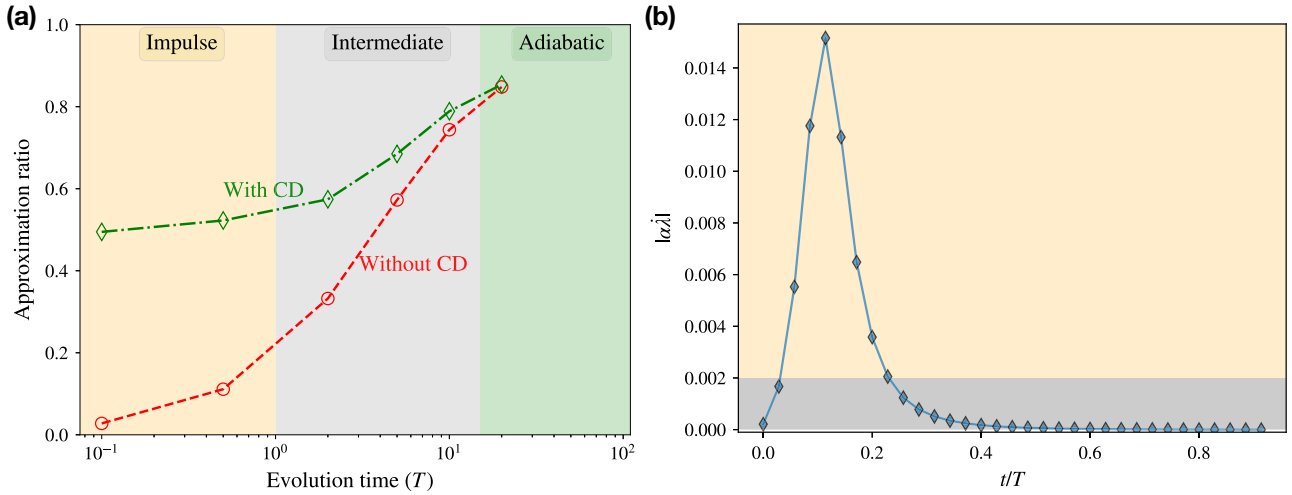

FIG. 1. For the purely quantum case, we present the simulation (a) of the trotterized evolution in three regimes under an adiabatic Hamiltonian and an adiabatic Hamiltonian with CD assistance. This simulation was conducted using the Qiskit local QASM simulator [43], employing a large number of shots (10^5) and a small step size (0.0075) to closely mimic exact time evolution. The regimes include the impulse regime (yellow), where CD dominates over adiabatic evolution; and the intermediate regime (gray), where the dominance of CD diminishes until reaching the adiabatic regime (green), in which CD is unnecessary and offers no advantage. (b) The magnitude of $\alpha\dot{\lambda}$, indicating the dominance of the CD term during evolution, is illustrated. The region where CD contributes most significantly is highlighted in yellow, while the region where it can be disregarded is in gray. The selection of this region is largely dependent on resource limitations. It is noteworthy that the peak in the magnitude of the CD term coincides with where nonadiabatic transitions are most likely to occur [44]. Both plots in this figure are theoretical results obtained from a selected 20-qubit portfolio optimization instance.

regime, the rate of change of the Hamiltonian is rapid, making the CD term more dominant than H_{ad}, as opposed to the adiabatic regime, where the CD term is inconsequential; see Fig. 1(a), where the regions are separated with respect to approximation ratio, defined as $(E_{\text{avg}} - E)/(E_{\text{avg}} - E_{\text{min}})$, with E_{avg} the average energy of the final Hamiltonian spectrum, E_{min} the ground state energy, and E the obtained energy. This definition will be discussed in Sec. III C. In addition, this observation is also justified by the fact that the two terms in Eq. (5), namely $H_{\text{ad}}(t)$ and A_λ, shift their relevance depending on their coefficients. Thus, in the impulse regime, where $|\lambda| \ll |\dot{\lambda}|$, the CD term (A_λ) dominates and the evolution is described by

$$H(t) \approx \dot{\lambda}(t) A_\lambda. \tag{10}$$

In our proposal, we work in the impulse regime; see Fig. 1(b) In addition to that, out of $N = T/dt$ Trotter steps, we only consider $p = N - q$ steps where the CD terms have significant contributions. The remaining q Trotter steps can be neglected with minimal impact on the quality of the final solution. In Fig. 2(a), we present a scheme of the purely quantum method named DCQO (CD only), where the dominance of the CD term during the trotterized evolution is highlighted based on the magnitude of $\alpha\dot{\lambda}$. For simplicity, we refer to this method just as *CD only*.

Furthermore, we implement a gate reduction strategy based on a user-defined threshold on gate angles. If the angle associated with a single- or multiqubit rotation is smaller than this threshold, the corresponding gate is not applied. This reduction is advantageous because the algorithm would only apply gates that are relevant enough to modify the state of the system up to a certain resolution given by the threshold. The reasonable threshold chosen for this study is a gate angle of less than 0.1. In addition, we note that for the IonQ hardware, the smallest gate angle their devices can accurately identify is 0.00628 [47]. Setting such a threshold aligns well with the principle of operating within the impulse regime, elucidating why the omission of certain gates is effective in this scenario.

We observe that one could further reduce the total gate counts by noting that the significance of the CD terms is mostly concentrated near the minimum gap, Δ_{min}. This region is where the majority of nonadiabatic transitions occur. Thus, it is essential to only apply the CD terms at these critical points, while determining the location of the minimum gap in advance poses a challenge. However, some recent proposals on adiabatic spectroscopy might be of help [48]. We can therefore pinpoint those Trotter steps where the CD term dominates, as in Fig. 1(b), and eliminate the quantum gates where the CD term has minimal impact, as in Fig. 2(a), ensuring efficiency without losing performance.

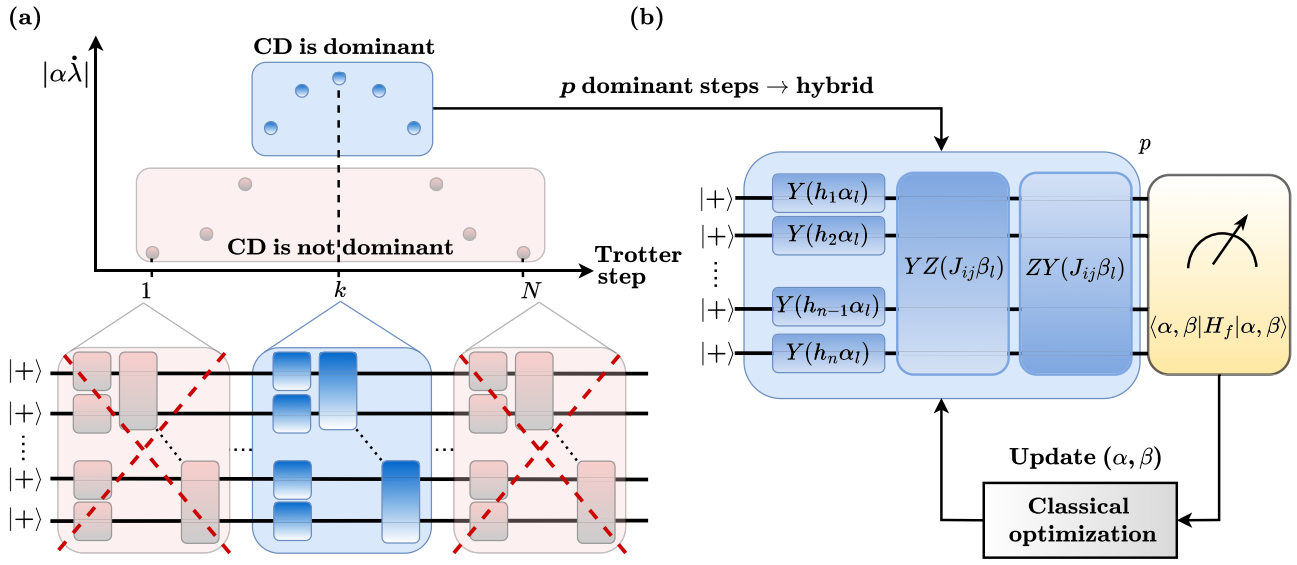

FIG. 2. (a) Purely quantum DCQO circuit. In the upper part, a qualitative behavior of $|\alpha\dot{\lambda}|$ is presented, which quantifies the dominance of the CD term, as in Fig. 1. The region where it is dominant (light blue) and where it is not (light red) is shown and mapped to a quantum circuit in the bottom part. Specifically, the first (kth) and last (Nth) steps are shown. The circuit blocks coming from the first and last steps are not applied since CD is not dominant, whereas the one coming from kth step is applied. The one-qubit blocks represent R_y rotations and the two-qubit ones represent R_{zy} followed by R_{yz}. (b) Hybrid DCQO ansatz, which has p layers that mimic p points from the dominant region of the CD term and assigns two parameters per layer, one for single-qubit rotations (α) and another one for two-qubit rotations (β). Afterwards, the usual hybrid procedure is applied to optimize the cost function $\langle \boldsymbol{\alpha}, \boldsymbol{\beta}|H_f|\boldsymbol{\alpha}, \boldsymbol{\beta}\rangle$. Both circuits are initialized by the equal superposition of states, continued by the $Y + YZ + ZY$ operator blocks obtained by the digitization of the CD term for the Ising type of Hamiltonians.

TABLE I. Scaling of the number of single-qubit gates (SQG) and two-qubit gates (TQG) with respect to the number of qubits n and trotter steps N.

Method	SQG	TQG
Without CD	$2nN$	$Nn(n-1)/2$
With CD	$3nN$	$3Nn(n-1)/2$
CD only	nN	$Nn(n-1)$

B. Hybrid-DCQO

We now propose the hybrid version of DCQO, which we call h-DCQO. This algorithm differs from the previously introduced hybrid algorithm DC-QAOA [22] in two respects. First, an h-DCQO ansatz is composed of only the CD term $\dot{\lambda}A_\lambda$, as opposed to the three terms in DC-QAOA (10). Second, in h-DCQO we consider one parameter for each CD term. A similar approach has already shown interesting results on other use cases such as protein folding [21]. Following the philosophy of variational quantum algorithms, the PQC generated from the h-DCQO ansatz is variationally optimized to minimize the expectation value.

As shown in Fig. 2(b), an h-DCQO ansatz is parameterized by α and β, which correspond to the one- and two-body Hamiltonian terms, respectively, of the first-order NC expansion of A_λ. Although both QAOA and h-DCQO ansatz are parameterized by the same number of parameters, the estimated two-qubit gate count for one layer of h-DCQO is double that of QAOA, which is $pN(N-1)/2$; see Table I. In spite of this increased gate count per layer, h-DCQO outperforms QAOA by reducing the required number of layers for a target performance, as we demonstrate empirically in the next section.

C. Model parameters and performance metric

We use Markowitz's model for portfolio optimization in our demonstrations. The coefficients of the problem Hamiltonian (3) are calculated from the unconstrained formulation (2), with the Lagrange multipliers $\theta_1 = 1$, $\theta_2 = 0.5$, and $\theta_3 = 2$, and historical stock prices. The values of the Lagrange multipliers are chosen by the user depending on how they emphasize the relative importance of the returns, risk and budget. We obtained real data associated with 20 assets, denoted by AAPL, JPM, JNJ, AMZN, PG, XOM, BA, DD, T, NEE, AMT, UPS, HD, PFE, NVDA, MSFT, GILD, GM, BRK-B, and LMT, for the period between June 6, 2022 and January 1, 2023 using the `YahooProvider` class from the Qiskit framework [43]. In addition, we choose a reasonable value for the budget, $B = n/2$.

To quantify the performance of an algorithm for solving the portfolio optimization problem, we propose a "mean-based" definition of the approximation ratio. Any quantum algorithm will give as output a state with energy E, where the optimal energy is $E_{\min}$. It is common to use a definition of the approximation ratio based on the spectral width $(E_{\max} - E_{\min})$ of the problem Hamiltonian [18], where the assumed worst-case solution corresponds to the maximum eigenenergy $E_{\max}$. In practice, the worst case is obtaining the average energy E_{avg} over a set of states where the solution is searched. Based on this, we modify the definition of the approximation ratio to

$$r^{(\text{avg})} \equiv \frac{E_{\text{avg}} - E}{E_{\text{avg}} - E_{\min}}. \tag{11}$$

We call this the mean-based approximation ratio and report our results using this metric. In particular, if the search set is an equal superposition of states $|+\rangle^{\otimes n}$)i.e., randomly guessing any of the 2^n solutions with equal probability), E_{avg} is zero, and our expression reduces to $r^{(\text{avg})} = E/E_{\min}$. We can use different sets (e.g., set of feasible solutions), described by different values of E_{avg}, which establishes a reference for the worst case.

IV. IMPLEMENTATION AND RESULTS

We performed our circuit simulations using the Qiskit local QASM simulator [43], all the hardware implementations using BraKet SDK from AWS [49]. For the hybrid algorithms, we leverage The Matter Lab's Tequila framework [50] and implement an expectation value minimization routine using COBYLA as the optimization of choice and limiting to 200 iterations. Given the all-to-all connectivity of the problem at hand, we select IonQ's trapped-ion device Aria 1, which has 25 fully connected qubits and is commercially available through AWS's BraKet service [49]. The calibration at the time of the demonstration can be found in Appendix C. Furthermore, we use Table I to choose the number of steps for purely quantum algorithms, as well as the number of layers for hybrid classical-quantum algorithms.

A. Pure DCQO solutions

In order to validate the DCQO (CD only) method, we examined 1000 random instances of the Ising spin-glass model with all-to-all interaction corresponding to the Hamiltonian in Eq. (3). The coefficients h_i and J_{ij} are drawn from a normal distribution with a mean of 0 and a variance of 1. We gathered results for systems of up to 20 qubits and compared the approximation ratios (11) among the three purely quantum methods previously mentioned. In Fig. 3, we observe the advantage of using DCQO (CD only) for fixed circuit depths. As expected, the addition of the CD term significantly improves the performance; in particular, a two-fold increase of the approximation ratio is observed. The comparable performance of DCQO with and without the adiabatic Hamiltonian term also justifies the use of just the CD Hamiltonian in our further studies. Furthermore, working with CD only not only results

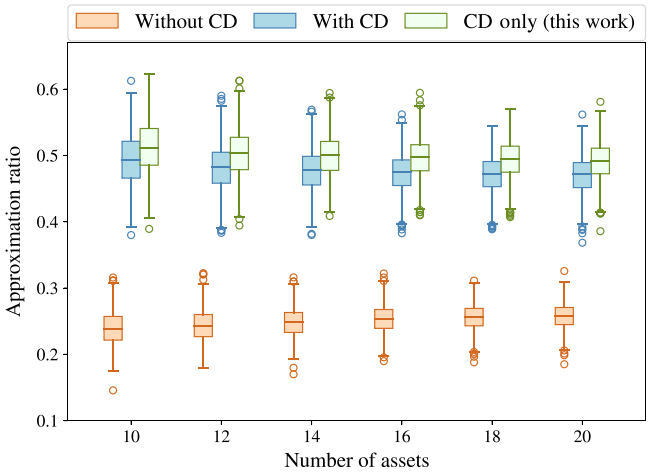

FIG. 3. Comparison of the performance of three methods: finite-time digitized adiabatic evolution (*without CD*), inclusive of the first-order CD (*with CD*), and the first-order CD term exclusively (*CD only*). The data represents the approximation ratio for 1000 randomly generated Ising spin-glass instances [refer to Eq. (3)] with 10 to 20 assets. The results were derived using a step size of $dt = 0.1$, with 12 Trotter steps in the without CD case, 4 steps in the with CD case, and 6 steps in the CD only case, ensuring consistent circuit depth across all scenarios; see Table I for reference.

in shallower circuits, making it more suitable for experiments, but also, in ideal simulations with trotterized evolution, having fewer noncommuting terms helps to reduce digitization error. This explains the slight improvement in performance observed with CD only, as shown in Fig. 3. In addition, it is important to note that the number of Trotter steps varies between using CD only and with CD to maintain the same circuit depth. This variation could also contribute to a slight shift in performance.

We present simulation results for a 20-asset portfolio optimization problem in Fig. 4(a). In this figure, we observe that seven steps of DCQO, which reduce to one step after gate cutoff, result in performance as good as 80 steps of finite-time digitized adiabatic evolution, since $r^{(\mathrm{avg})} \approx 0.54$ for both cases. This means a 40-fold reduction in the circuit depth as the depth-per-step for DCQO (CD only) is twice that for finite-time digitized adiabatic evolution; see Table I for reference.

For the hardware implementations, a set of modifications are performed to the logical representation in Fig. 2. In particular, transpilation of logical gates to IonQ's native gates is performed using their native gate set $\{GPi, GPi2, MS\}$; see Appendix B for details. In addition, the error mitigation (EM) strategy of debiasing is added to reduce systematic errors on the hardware [51]. The performance gap between the simulated and hardware executions (without EM) is evident from Fig. 4(b). With EM, we are able to boost the performance to $r^{(\mathrm{avg})} = 0.50$, which is comparable to the simulated result. This highlights the fact that for sufficiently low-depth circuits, like that of DCQO (CD only), debiased results from IonQ hardware are quite faithful to the ideal solutions.

B. Hybrid DCQO solutions

In Fig. 5(a) we observe that a five-layer QAOA performs as well as a one-layer h-DCQO. Although the depth per layer for h-DCQO is twice that of QAOA, this five-fold reduction in the required number of layers ultimately leads

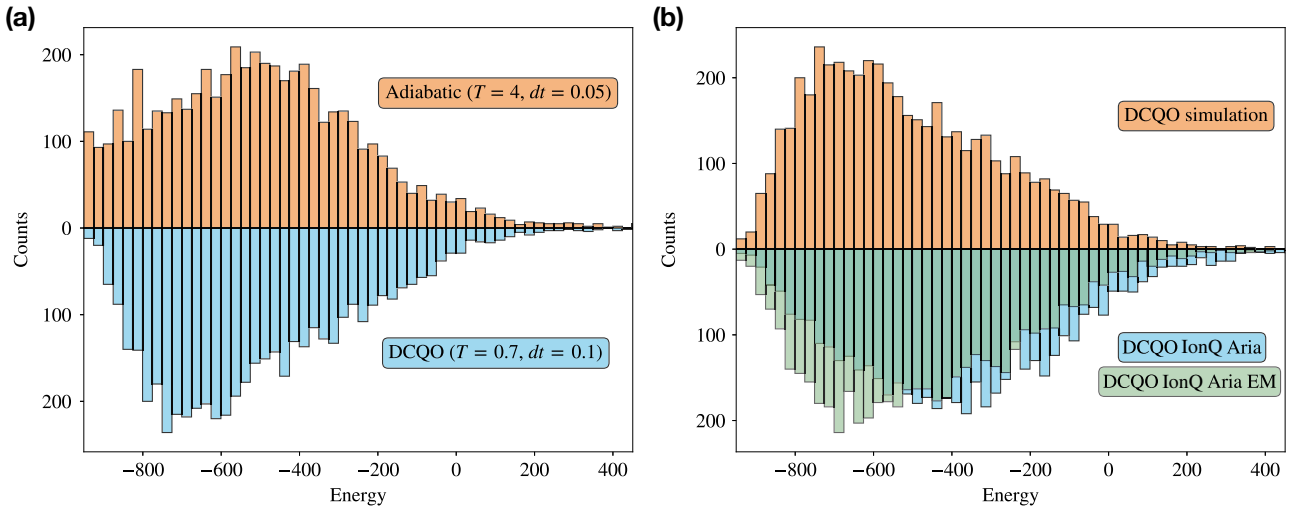

FIG. 4. (a) Energy distributions of the final state after applying DCQO (7 Trotter steps) and adiabatic (80 Trotter steps). The approximation ratios are 0.54 and 0.53, respectively. (b) Energy distributions of the final state after applying DCQO in IonQ's 25-qubit device Aria through AWS with and without error mitigation (debias) and the simulation results. The approximation ratios are 0.50, 0.40, and 0.54, respectively. The number of shots was 5000 in both cases, which was the maximum supported by the hardware at the time of use. The gate cutoff threshold was set to 0.1.

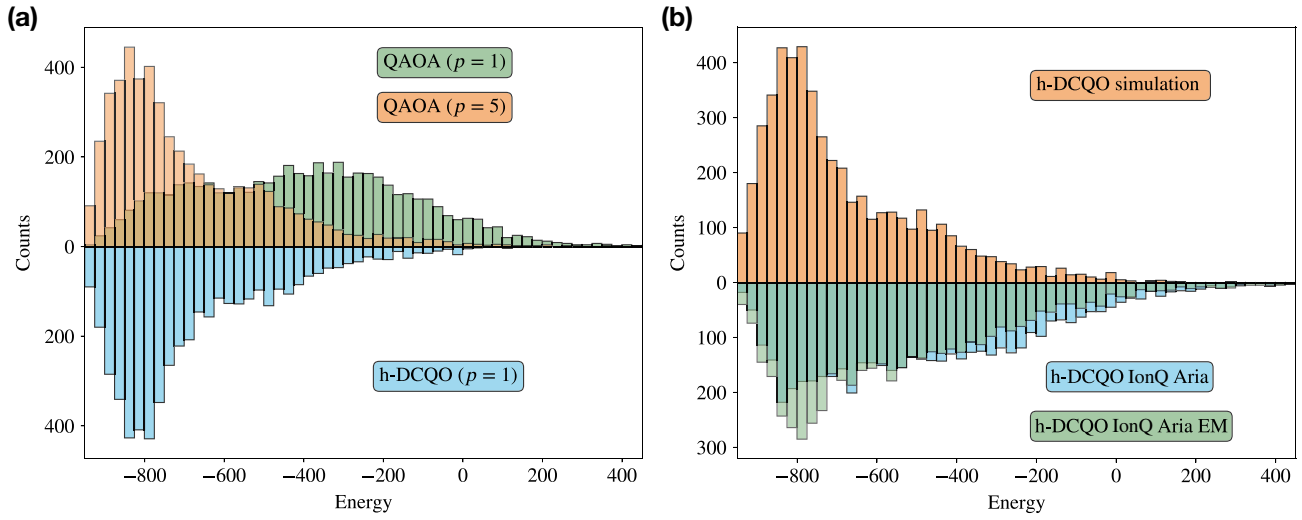

FIG. 5. (a) Energy distributions of the final state after applying h-DCQO ($p = 1$ layer) and QAOA ($p = 1$ and $p = 5$ layers) at the optimal parameters. The approximation ratios are 0.72, 0.41, and 0.73, respectively. (b) Energy distributions of the final state after applying h-DCQO at the optimal parameters on IonQ's 25-qubit Aria device through AWS with and without error mitigation (debias), and the simulation results. The approximation ratios are 0.58, 0.52, and 0.73, respectively. The number of shots was 5000 in both cases, which was the maximum supported by the hardware at the time of usage.

to a 2.5-fold reduction in circuit depth for h-DCQO as compared to QAOA; see Table I for reference. The reduction in the number of layers also leads to a five-fold reduction in the number of parameters, which, consequently, makes the variational training faster. A one-layer QAOA, on the other hand, has much worse performance. Thus our h-DCQO algorithm is able to compress the required circuit depth for a target performance of $r^{(\mathrm{avg})} = 0.72$ for the portfolio optimization problem. The circuit-depth compression by h-DCQO is relevant for being able not only to implement any algorithm within the coherence time of the hardware but also to decrease the cumulative effect of hardware noise.

In Fig. 5(b), we present results from our hardware implementation of h-DCQO and deliberately ignore QAOA due to its poor performance at low depth. After performing the parameter learning locally, we execute the final circuit on IonQ's hardware. While the mitigated results boost the section coinciding with the highest success probability states, the tail of the distribution is increased and noise mitigation does not seem to remove this, making the approximation ratio $r^{(\mathrm{avg})} = 0.58$. The existence of a significant gap between mitigated and unmitigated results, in the case of h-DCQO, is due to the fact that we do not apply the gate cutoff threshold for h-DCQO, since the variationally obtained angles were larger in magnitude, yielding deeper circuits than in DCQO. Thus, further noise suppression techniques are needed in order to close the gap between simulation and demonstration (with EM) for the h-DCQO algorithm.

V. CONCLUSIONS

In this work, we demonstrated a drastic reduction in the quantum computational resources when tackling the portfolio optimization problem without compromising the solution quality. We achieved this important result by operating counterdiabatic dynamics in the fast-evolution or impulse regime, where the CD terms are dominant. In addition, we outlined how to choose the corresponding Trotter steps so that the magnitude of the CD terms makes a significant contribution. Using the approximation ratio as a metric, we illustrated that both the pure quantum approach (DCQO) and the hybrid approach (h-DCQO) outperform digitized adiabatic quantum protocols and QAOA. In this manner, we achieved a substantial reduction in the circuit complexity while maintaining a similar solution accuracy. Furthermore, we performed a cloud-based demonstration of our proposal on a 20-qubit trapped-ion quantum computer, using real-world portfolio data involving 20 assets. Finally, we validated the performance of our methods on a general Ising spin-glass problem with all-to-all interactions, underscoring their applicability to other combinatorial optimization problems.

As a final remark, we believe it is crucial to develop powerful methods to compress digital algorithms within the coherence time of available quantum computers. Furthermore, adopting a digital-analog approach may offer an intriguing new direction towards enhancing the practical utility of quantum computing in the noisy intermediate-scale quantum era.

ACKNOWLEDGMENTS

We are grateful for the open access of Amazon Braket SDK and its associated service enabling access to IonQ's Aria device, as well as The Matter Lab's Tequila frameworks. The authors acknowledge Pranav Chandarana and Francisco Albarrán-Arriagada for useful discussions and feedback.

APPENDIX A: EXPLICIT FORM OF THE ISING COEFFICIENTS

The maximization of Eq. (2) is equivalent to the minimization of the function

$$F = -\theta_1 \sum_{i=1}^{n} x_i e_i + \theta_2 \sum_{i,j=1}^{n} x_i x_j c_{ij} + \theta_3 \left(\sum_{i=1}^{n} x_i - B \right)^2 \tag{A1}$$

$$= -\sum_{i=1}^{n} \eta_i x_i + \sum_{i,j=1}^{n} \kappa_{ij} x_i x_j + C, \tag{A2}$$

where $\eta_i = \theta_1 e_i + 2\theta_3 B$ and $\kappa_{ij} = \theta_2 c_{ij} + \theta_3$ can be found directly from the expansion of the constraint term, and C is a nonrelevant offset. Now, using $x_i = \frac{1}{2}(1 - Z_i)$,

$$F = \sum_{i=1}^{n} h_i Z_i + \sum_{i<j} J_{ij} Z_i Z_j + C, \tag{A3}$$

where C is again a nonrelevant offset and the Ising coefficients are $h_i = \frac{1}{2}\left(\eta_i - \sum_j \kappa_{ij}\right)$ and $J_{ij} = \frac{1}{2}\kappa_{ij}$. Finally, Eq. (A3) is equivalent to the Ising formulation of Eq. (3).

APPENDIX B: ANALYTICAL CALCULATION OF CD COEFFICIENT FROM THE FIRST ORDER NESTED COMMUTATOR

The first-order CD coefficient can be computed exactly for any Hamiltonian, provided the first two nested commutators $\mathcal{O}_1 = [H_{\text{ad}}, \partial_\lambda H_{\text{ad}}]$ and $\mathcal{O}_2 = [H_{\text{ad}}, \mathcal{O}_1]$, that is, $\alpha_1 = -\Gamma_1/\Gamma_2$, where $\Gamma_k = \|\mathcal{O}_k\|^2$ such that the CD term takes the form

$$H_{\text{CD}} = i\dot{\lambda}\alpha_1 \mathcal{O}_1. \tag{B1}$$

For the case of the Hamiltonians used in Eq. (4),

$$\begin{aligned} \mathcal{O}_1 &= \left[-\sum_i X_i, \sum_k h_k Z_k + \sum_{kl} J_{k<l} Z_k Z_k \right] \\ &= 2i \sum_i h_i Y_i + 2i \sum_{i<j} J_{ij} \left(Y_i Z_j + Z_i Y_j \right). \end{aligned} \tag{B2}$$

Therefore, Γ_1 is given by

$$\Gamma_1 = 4 \sum_i h_i^2 + 4 \sum_{i<j} J_{ij}^2. \tag{B3}$$

Now, we need to calculate the denominator of the equation for α_1, which follows from $\mathcal{O}_2 = (1-\lambda)[H_i, \mathcal{O}_1] + \lambda\left[H_f, \mathcal{O}_1\right]$. Then, we need to evaluate two different commutators.
The first one can be written as

$$[H_i, \mathcal{O}_1] = 4 \sum_i h_i Z_i + 8 \sum_{i<j} J_{ij} Z_i Z_j - 8 \sum_{i<j} J_{ij} Y_i Y_j \tag{B4}$$

whereas the second one is

$$\left[H_f, \mathcal{O}_1\right] = 4 \sum_i \left(h_i^2 + \sum_j J_{ij}^2 \right) X_i + 8 \sum_{i \neq j} h_i J_{ij} X_i Z_j + 8 \sum_{i<j<k} \left(J_{ij} J_{ij} X_i Z_j Z_k + J_{ij} J_{jk} Z_i X_j Z_k + J_{ik} J_{jk} Z_i Z_j X_k \right). \tag{B5}$$

It follows that

$$\Gamma_2 = 16(1-\lambda)^2 \left(\sum_i h_i^2 + 4 \sum_{i \neq j} J_{ij}^2 \right) + 16\lambda^2 \left(\sum_i h_i^4 + \sum_{i \neq j} J_{ij}^4 + 6 \sum_{i \neq j}^{1} h_i^2 J_{ij}^2 + 6 \sum_{i<j<k} \left(J_{ij}^2 J_{ik}^2 + J_{ij}^2 J_{jk}^2 + J_{ik}^2 J_{jk}^2 \right) \right). \tag{B6}$$

Hence, the CD term is given by

$$H_{\text{CD}} = -2\dot{\lambda}\alpha_1 \left(\sum_i h_i Y_i + \sum_{i<j} J_{ij} \left(Y_i Z_j + Z_i Y_j \right) \right), \quad \text{where } \alpha_1 = -\frac{1}{4} \frac{\sum_i h_i^2 + \sum_{i<j} J_{ij}^2}{R(t)}, \tag{B7}$$

and

$$R(t) = (1-\lambda)^2\left(\sum_i h_i^2 + 4\sum_{i\neq j} J_{ij}^2\right) + \lambda^2\left(\sum_i h_i^4 + \sum_{i\neq j} J_{ij}^4 + 6\sum_{i\neq j}^{1} h_i^2 J_{ij}^2 + 6\sum_{i<j<k}\left(J_{ij}^2 J_{ik}^2 + J_{ij}^2 J_{jk}^2 + J_{ik}^2 J_{jk}^2\right)\right). \quad \text{(B8)}$$

APPENDIX C: IONQ ARIA 1 CALIBRATION AT THE TIME OF THE DEMONSTRATION

```
'connectivity':'all-to-all',
'qubits': 25,
'fidelity': {'1q': {'mean': 0.9997},
 '2q': {'mean': 0.9878},
 'spam': {'mean': 0.9943}},
'timing': {'t1': 100 s,
 't2': 1 s,
 '1q': 0.000135 s,
 '2q': 0.000a6 s,
 'readout': 0.0003 s,
 'reset': 2e-05 s},
'date': 1688591883
```

APPENDIX D: IONQ NATIVE GATE IMPLEMENTATION ON AMAZON BRAKET

The transpilation process was entirely done by finding optimal ways to translate the logical gate unitaries to their hardware-specific representation. In the case of the IonQ device, it is based on their reference material [52] where they represent their native *GPi*, *GPi*2, and *MS* gate unitaries.

Logical to physical gates using single-qubit operations were translated by looking for the appropriate parameter definition following the specification

$$GPi(\phi) = \begin{bmatrix} 0 & e^{-i\phi} \\ e^{i\phi} & 0 \end{bmatrix}, \qquad GPi2(\phi) = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & -ie^{-i\phi} \\ -ie^{i\phi} & 1 \end{bmatrix},$$

$$MS(\phi_0,\phi_1,\theta) = \begin{bmatrix} \cos\frac{\theta}{2} & 0 & 0 & -ie^{-i(\phi_0+\phi_1)}\sin\frac{\theta}{2} \\ 0 & \cos\frac{\theta}{2} & -ie^{-i(\phi_0-\phi_1)}\sin\frac{\theta}{2} & 0 \\ 0 & -ie^{i(\phi_0-\phi_1)}\sin\frac{\theta}{2} & \cos\frac{\theta}{2} & 0 \\ -ie^{i(\phi_0+\phi_1)}\sin\frac{\theta}{2} & 0 & 0 & \cos\frac{\theta}{2} \end{bmatrix},$$

which produces the replacements shown in Table II. Two-qubit gates use the native Mølmer-Sørenson gate as the basis of the translation. Depending on the angle of rotation, different choices of the MS gate are selected based on their proximity to the extremes of a 2π circle. We used the definition of the partially entangling MS gate as described by IonQ, which requires three parameters in all cases, where the third parameter must be $\theta/2\pi \in [0, \frac{1}{4}]$, being the maximum value equal to the fully entangled MS gate implementation. Table III shows the implementation of the $R_{yy}(\theta)$ gate depending on the third parameter restriction on the angle being selected with respect to the 2π reference. Taking this as the basis, $R_{yy}(\theta)$ gates can be easily translated into different two-qubit gates needed for QAOA or hybrid DCQO by simply surrounding this action with the corresponding single-gate rotations per qubit in the gate implementation. Along with these translations, Amazon Braket SDK allows for debias for systematic error mitigation [51] by simply invoking it during the job submission.

TABLE II. Transpilation of some important gates that are present in the Hamiltonians we worked with. Specifically, we implemented in hardware the gates H, $R_y(\theta)$, $R_{yz}(\theta)$, and $R_{zy}(\theta)$ as shown in this table.

Logical gate	Native decomposition
H	$GPi2(\pi/2) \times GPi(0)$
$R_z(\theta)$	$GPi(0) \times GPi(\theta/2)$
$R_y(\theta)$	$GPi2(\pi) \times GPi(\theta/2) \times GPi2(\pi)$
$R_{zz}(\theta)$	$(GPi2(\pi) \otimes GPi2(\pi))R_{yy}(\theta)(GPi2(0) \otimes GPi2(0))$
$R_{zy}(\theta)$	$(GPi2(\pi) \otimes I)R_{yy}(\theta)(GPi2(0) \otimes I)$
$R_{yz}(\theta)$	$(I \otimes GPi2(\pi))R_{yy}(\theta)(I \otimes GPi2(0))$

TABLE III. Transpilation of the gate $R_{yy}(\theta)$ in terms of IonQ native gates for four different ranges of θ.

Angle	Native gate
$\theta \leq \frac{\pi}{2}$	$MS(\pi/2,\pi/2,\theta)$
$\frac{\pi}{2} < \theta \leq \pi$	$(GPi(\pi/2)\otimes GPi(\pi/2)) \times MS(3\pi/2,\pi/2,\pi-\theta)$
$\pi < \theta \leq \frac{3\pi}{4}$	$(GPi(\pi/2)\otimes GPi(\pi/2)) \times MS(\pi/2,\pi/2,\theta-\pi)$
$\pi < \theta < \frac{3\pi}{2}$	$MS(3\pi/2,\pi/2,2\pi-\theta)$

The main objective of this simple error mitigation technique was to avoid overcompensating by additional and more complex techniques.

[1] P. W. Shor, in *Proceedings 35th Annual Symposium on Foundations of Computer Science* (IEEE, Santa Fe, NM, USA, 1994), p. 124.

[2] C. Gidney and M. Ekerå, How to factor 2048 bit RSA integers in 8 h using 20 million noisy qubits, Quantum **5**, 433 (2021).

[3] K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, and T. Menke *et al.*, Noisy intermediate-scale quantum algorithms, Rev. Mod. Phys. **94**, 015004 (2022).

[4] R. Orús, S. Mugel, and E. Lizaso, Quantum computing for finance: Overview and prospects, Rev. Phys. **4**, 100028 (2019).

[5] S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan, Quantum computational chemistry, Rev. Mod. Phys. **92**, 015003 (2020).

[6] E. Farhi, J. Goldstone, and S. Gutmann, A quantum approximate optimization algorithm, arXiv:1411.4028.

[7] L. Zhou, S.-T. Wang, S. Choi, H. Pichler, and M. D. Lukin, Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices, Phys. Rev. X **10**, 021067 (2020).

[8] E. Farhi, J. Goldstone, S. Gutmann, and L. Zhou, The quantum approximate optimization algorithm and the Sherrington-Kirkpatrick model at infinite size, Quantum **6**, 759 (2022).

[9] S. Boulebnane and A. Montanaro, Solving boolean satisfiability problems with the quantum approximate optimization algorithm, PRX Quantum **5**, 030348 (2024).

[10] M. P. Harrigan, K. J. Sung, M. Neeley, K. J. Satzinger, F. Arute, K. Arya, J. Atalaya, J. C. Bardin, R. Barends, and S. Boixo *et al.*, Quantum approximate optimization of non-planar graph problems on a planar superconducting processor, Nat. Phys. **17**, 332 (2021).

[11] S. Wang, E. Fontana, M. Cerezo, K. Sharma, A. Sone, L. Cincio, and P. J. Coles, Noise-induced barren plateaus in variational quantum algorithms, Nat. Commun. **12**, 6961 (2021).

[12] M. Streif and M. Leib, Training the quantum approximate optimization algorithm without access to a quantum processing unit, Quantum Sci. Technol. **5**, 034008 (2020).

[13] M. Larocca, P. Czarnik, K. Sharma, G. Muraleedharan, P. J. Coles, and M. Cerezo, Diagnosing barren plateaus with tools from quantum optimal control, Quantum **6**, 824 (2022).

[14] S. H. Sack and D. J. Egger, Large-scale quantum approximate optimization on non-planar graphs with machine learning noise mitigation, Phys. Rev. Res. **6**, 013223 (2024).

[15] D. Herman, C. Googin, X. Liu, Y. Sun, A. Galda, I. Safro, M. Pistoia, and Y. Alexeev, Quantum computing for finance, Nat. Rev. Phys. **5**, 1 (2023).

[16] D. J. Egger, J. Mareček, and S. Woerner, Warm-starting quantum optimization, Quantum **5**, 479 (2021).

[17] S. Brandhofer, D. Braun, V. Dehn, G. Hellstern, M. Hüls, Y. Ji, I. Polian, A. S. Bhatia, and T. Wellens, Benchmarking the performance of portfolio optimization with QAOA, Quantum Inf. Process. **22**, 25 (2022).

[18] Z. He, R. Shaydulin, S. Chakrabarti, D. Herman, C. Li, Y. Sun, and M. Pistoia, Alignment between initial state and mixer improves QAOA performance for constrained portfolio optimization, npj Quantum Inf. **9**, 121 (2023).

[19] N. N. Hegade, K. Paul, Y. Ding, M. Sanz, F. Albarrán-Arriagada, E. Solano, and X. Chen, Shortcuts to adiabaticity in digitized adiabatic quantum computing, Phys. Rev. Appl. **15**, 024038 (2021).

[20] I. Čepaitė, A. Polkovnikov, A. J. Daley, and C. W. Duncan, Counterdiabatic optimized local driving, PRX Quantum **4**, 010312 (2023).

[21] P. Chandarana, N. N. Hegade, I. Montalban, E. Solano, and X. Chen, Digitized counterdiabatic quantum algorithm for protein folding, Phys. Rev. Appl. **20**, 014024 (2023).

[22] P. Chandarana, N. N. Hegade, K. Paul, F. Albarrán-Arriagada, E. Solano, A. Del Campo, and X. Chen, Digitized-counterdiabatic quantum approximate optimization algorithm, Phys. Rev. Res. **4**, 013141 (2022).

[23] M. Vizzuso, G. Passarelli, G. Cantele, and P. Lucignano, Convergence of digitized-counterdiabatic QAOA: Circuit depth versus free parameters, New J. Phys. **26**, 013002 (2024).

[24] Q.-M. Ding, Y.-M. Huang, and X. Yuan, Molecular docking via quantum approximate optimization algorithm, Phys. Rev. Appl. **21**, 034036 (2024).

[25] H. Markowitz, Portfolio selection, J. Finance. **7**, 77 (1952).

[26] H. Kellerer, R. Mansini, and M. G. Speranza, Selecting portfolios with fixed costs and minimum transaction lots, Ann. Oper. Res. **99**, 287 (2000).

[27] N. Elsokkary, F. S. Khan, D. La Torre, T. S. Humble, and J. Gottlieb, *Financial portfolio management using d-wave quantum optimizer: The case of Abu Dhabi Securities Exchange*, Tech. Rep. (Oak Ridge National Lab.(ORNL), Oak Ridge, TN (United States), 2017).

[28] M. Parizy, P. Sadowski, and N. Togawa, in *2022 IEEE 35th International System-on-Chip Conference (SOCC)* (IEEE, Belfast, United Kingdom, 2022), p. 1.

[29] M. Lopez de Prado, Generalized optimal trading trajectories: A financial quantum computing application, Available at SSRN, 2575184 (2015).

[30] G. Rosenberg, P. Haghnegahdar, P. Goddard, P. Carr, K. Wu, and M. L. De Prado, in *Proceedings of the 8th Workshop on High Performance Computational Finance* (2015), p. 1.

[31] S. Yarkoni, E. Raponi, T. Bäck, and S. Schmitt, Quantum annealing for industry applications: Introduction and review, Rep. Prog. Phys. **85**, (2022).

[32] M. Demirplak and S. A. Rice, Adiabatic population transfer with control fields, J. Phys. Chem. A **107**, 9937 (2003).

[33] M. V. Berry, Transitionless quantum driving, J. Phys. A: Math. Theor. **42**, 365303 (2009).

[34] A. del Campo, Shortcuts to adiabaticity by counterdiabatic driving, Phys. Rev. Lett. **111**, 100502 (2013).

[35] D. Sels and A. Polkovnikov, Minimizing irreversible losses in quantum systems by local counterdiabatic driving, Proc. Natl. Acad. Sci. **114**, E3909 (2017).

[36] P. W. Claeys, M. Pandey, D. Sels, and A. Polkovnikov, Floquet-engineering counterdiabatic protocols in quantum many-body systems, Phys. Rev. Lett. **123**, 090602 (2019).
[37] T. Hatomura and K. Takahashi, Controlling and exploring quantum systems by algebraic expression of adiabatic gauge potential, Phys. Rev. A **103**, 012220 (2021).
[38] K. Takahashi and A. del Campo, Shortcuts to adiabaticity in Krylov space, Phys. Rev. X **14**, 011032 (2024).
[39] L. K. Kovalsky, F. A. Calderon-Vargas, M. D. Grace, A. B. Magann, J. B. Larsen, A. D. Baczewski, and M. Sarovar, Self-healing of trotter error in digital adiabatic state preparation, Phys. Rev. Lett. **131**, 060602 (2023).
[40] Y.-A. Chen, A. M. Childs, M. Hafezi, Z. Jiang, H. Kim, and Y. Xu, Efficient product formulas for commutators and applications to quantum simulation, Phys. Rev. Res. **4**, 013191 (2022).
[41] J. Wurtz and P. J. Love, Counterdiabaticity and the quantum approximate optimization algorithm, Quantum **6**, 635 (2022).
[42] J. Yao, L. Lin, and M. Bukov, Reinforcement learning for many-body ground-state preparation inspired by counterdiabatic driving, Phys. Rev. X **11**, 031070 (2021).
[43] Qiskit contributors, Qiskit: An open-source framework for quantum computing (2023).
[44] M. V. Berry, Transitionless quantum driving, J. Phys. A: Math. Theor. **42**, 365303 (2009).
[45] Q. Xie, K. Seki, and S. Yunoki, Variational counterdiabatic driving of the Hubbard model for ground-state preparation, Phys. Rev. B **106**, 155153 (2022).
[46] E. Carolan, A. Kiely, and S. Campbell, Counterdiabatic control in the impulse regime, Phys. Rev. A **105**, 012605 (2022).
[47] Getting started with native gates, https://ionq.com/docs/getting-started-with-native-gates, accessed: 2023-08-22.
[48] B. F. Schiffer, J. Tura, and J. I. Cirac, Adiabatic spectroscopy and a variational quantum adiabatic algorithm, PRX Quantum **3**, 020347 (2022).
[49] Amazon Web Services, Amazon BraKet (2023).
[50] J. S. Kottmann, S. Alperin-Lea, T. Tamayo-Mendoza, A. Cervera-Lierta, C. Lavigne, T.-C. Yen, V. Verteletskyi, P. Schleich, A. Anand, M. Degroote, S. Chaney, M. Kesibi, N. G. Curnow, B. Solo, G. Tsilimigkounakis, C. Zendejas-Morales, A. F. Izmaylov, and A. Aspuru-Guzik, Tequila: A platform for rapid development of quantum algorithms, Quantum Sci. Technol. **6**, 024009 (2021).
[51] A. Maksymov, J. Nguyen, Y. Nam, and I. Markov, Enhancing quantum computer performance via symmetrization, arXiv:2301.07233 [quant-ph].
[52] https://ionq.com/docs/getting-started-with-native-gates.