# Efficient DCOO Algorithm within the Impulse Regime for Portfolio Optimization

Alejandro Gomez Cadavid, <sup>1</sup> Iraitz Montalban, <sup>1, 2</sup> Archismita Dalal, <sup>1</sup> Enrique Solano, <sup>1, \*</sup> and Narendra N. Hegade <sup>1, †</sup> <sup>1</sup> Kipu Quantum, Greifswalderstrasse 226, 10405 Berlin, Germany

<sup>2</sup>Department of Physics, University of the Basque Country UPV/EHU, Barrio Sarriena, s/n, 48940 Leioa, Biscay, Spain

We propose a faster digital quantum algorithm for portfolio optimization using the digitized-counterdiabatic quantum optimization (DCQO) 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 experimentally 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.

#### I. INTRODUCTION

Quantum computing aims at challenging the status quo of classical computation for hard-to-solve problems, as it was first demonstrated in Shor's quantum algorithm for prime factorization [1]. However, the hardware requirements raise valid concerns on 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 codesigned 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, among others [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]. QAOA is one of the hybrid classical-quantum approaches known as variational quantum algorithms (VQA), and it requires classical training to fit an ansatz or parametrized quantum circuit (POC) composed of discrete gates. Similarly to Shor's algorithm, it does suffer 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 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 con-

venience 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 previous results [19–24], as we explain in detail along this article. We rely on the concept of the impulse regime, which corresponds to the fast non-adiabatic 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 §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. Next in §III, we elaborate on our proposed algorithms of DCQO and h-DCQO and introduce our modified definition for approximation ratio. Then we present our results, from both simulation and experiment, and discuss them in §IV. Finally, we conclude with an outlook in §V.

#### 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 time steps. 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 will focus on a single-time step modality of this problem with Boolean asset investment.

<sup>\*</sup> enrique.solano@kipu-quantum.com

<sup>†</sup> narendra.hegade@kipu-quantum.com

Let us consider then, a single time step t and a budget B, which defines the number of assets that can be selected. The budget sets a constraint to the problem, which is described by the summation cost  $x_i$  of investing on each individual asset i. Ideally, the solution should not surpass this budget condition  $(\sum_i x_i \leq B)$ . Therefore, the problem is posed in the first stage as follows

$$\max_{x_{i} \in \{0,1\}} \sum_{i=1}^{n} x_{i} e_{i} - \theta \sum_{i,j=1}^{n} x_{i} x_{j} c_{ij}$$
s.t. 
$$\sum_{i=1}^{n} x_{i} \leq B.$$
(1)

Here,  $x_i \in \{0, 1\}$  is the mask associated with the selection of our set of assets, while  $\theta$  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

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

Here, the budget constraint has been integrated to the cost function and  $\theta_{1,2,3}$  are Lagrange multipliers, which decide the relevance of each term. In this way, one can modulate not only the risk but also the option of surpassing 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 be easily 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 to find the ground state of an Ising spin-glass Hamiltonian  $H_f$ , which we call the problem Hamiltonian,

$$H_{\rm f} = \sum_{i=1}^{n} h_i Z_i + \sum_{i < j} J_{ij} Z_i Z_j.$$
 (3)

Its ground state encodes the binary string, or strings in 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  $\left(\frac{|0\rangle+|1\rangle}{\sqrt{2}}\right)^{\otimes n}$ , towards the ground state of the problem Hamiltonian  $H_{\rm f}$ . This evolution is governed by the Hamiltonian

$$H_{\rm ad}(t) = [1 - \lambda(t)]H_{\rm i} + \lambda(t)H_{\rm f},\tag{4}$$

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. Along this article, we use the following scheduling function  $\lambda(t) = \sin^2\left[\frac{\pi}{2}\sin^2(\frac{\pi t}{2T})\right]$ , such that  $\lambda(0) = \lambda(T) = 0$ .

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 non-adiabatic 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 non-adiabatic transitions due to rapid evolution [32–34]. This driving is introduced as an extension of Eq. (4) and can be written in the form

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

Here,  $A_{\lambda}$  is known as the adiabatic gauge potential, responsible for suppressing the non-adiabatic excitations. The calculation of the gauge potential is computationally expensive since it requires knowledge of the instantaneous eigenspectrum of  $H_{\rm ad}$  [33]. However, taking the proposal from Ref. [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), i.e.

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

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

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

The detailed calculation of the analytical expression of  $\alpha_1$  is given in Appendix A.

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  $H(t) = \sum_k c_k(t)H_k$ . 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 expense 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 been introduced recently to realize digitized-counterdiabatic evolution [40].

# C. Hybrid approaches

Purely quantum approaches may often require long circuit depths coming 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, equivalent to  $H_{\rm f}$  and  $H_{\rm i}$  in Eq. (4), respectively. This algorithm has become in recent years 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, i.e.  $|+\rangle^{\otimes n}$ . Then, following alternating layers of the unitary associated to the problem Hamiltonian  $U_f(\gamma) = e^{-i\gamma H_i}$ , parameterized by  $\gamma$ , and a mixer unitary  $U_i(\beta) = e^{-i\beta H_i}$ , parameterized by  $\beta$ , the algorithm produces the following state

$$|\gamma,\beta\rangle = U_i(\beta_p)U_f(\gamma_p)\dots U_i(\beta_1)U_f(\gamma_1)|+\rangle.$$
 (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  $\alpha$ ,

$$|\alpha, \beta, \gamma\rangle = U_c(\alpha_p)U_i(\beta_p)U_f(\gamma_p)\dots$$

$$U_c(\alpha_1)U_i(\beta_1)U_f(\gamma_1)|+\rangle. \tag{10}$$

The extension of QAOA using CD protocols, known as DC-QAOA, was introduced in Refs. [22, 41, 42], which has been applied to various problems, e.g. 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 [43]. These regimes are characterized by the total evolution time, with short times lying in the impulse regime and long times falling in the adiabatic regime. In the impulse regime, the rate of change of the Hamiltonian is rapid, making the CD term more dominant than  $H_{\rm ad}$  as opposed to the adiabatic regime, where the CD term is inconsequential; see Fig. 1(a). Additionally, this observation is also justified by the fact that the two terms in Eq. (5), namely  $H_{\rm ad}(t)$  and  $A_{\lambda}$ , shift their relevance depending on their coefficients. Thus in the impulse regime, where  $|\lambda| << |\dot{\lambda}|$ , the CD term  $(A_{\lambda})$  dominates and the evolution is described by

$$H(t) \approx \dot{\lambda}(t)A_{\lambda}.$$
 (11)

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 DCQO method, where the dominance of the CD term during the Trotterized evolution is highlighted based on the magnitude of  $\alpha \dot{\lambda}$ .

Furthermore, we implement a gate reduction strategy based on a user-defined threshold on gate angles. If the angle associated with a single or multi-qubit rotation is smaller than this

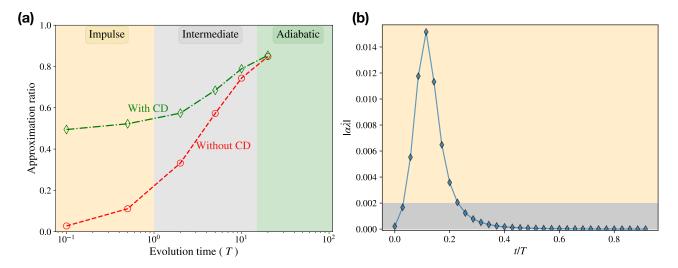


FIG. 1. For the purely quantum case, we show the simulated (a) three regimes of evolution under adiabatic Hamiltonian and adiabatic Hamiltonian + CD assistance. The impulse regime (yellow), where CD is dominant over Adiabatic, the intermediate regime (gray), where CD starts to lose dominance until the adiabatic regime (green), where CD is no longer needed nor offers any advantage. (b) Magnitude of  $\alpha\lambda$ , which measures the dominance of the CD term. An estimation of the region in which CD has the most important contribution is highlighted in yellow whereas the one that can be ignored is in grey. The selection of this region would mostly depend on the resource limitations. These results were obtained using a selected 20 qubit portfolio optimization instance.

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. Additionally, we note that for the IonQ hardware, the smallest gate angle their devices can accurately identify is 0.00628 [44]. 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.

As a remark, one could further reduce the total gate counts by noting that the significance of the CD terms is mostly concentrated near the minimum gap,  $\Delta_{min}$ . This region is where the majority of non-adiabatic 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 [45]. We can therefore pinpoint those Trotter steps where the CD term dominates and eliminate the quantum gates where the CD term has minimal impact, ensuring efficiency without losing performance.

#### B. Hybrid-DCQO

Now, we 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 aspects. First, a h-DCQO ansatz is composed of only the CD term  $\lambda A_{\lambda}$ , 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 already showed interesting results on other

use cases like 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), a h-DCQO ansatz is parameterized by  $\alpha$  and  $\beta$ , which correspond to the one- and two-body Hamiltonian terms, respectively, of the first order NC expansion of  $A_{\lambda}$ . 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 than that of QAOA, which is pN(N-1)/2. 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 experiments. 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, labeled 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 06.06.2022 and 01.01.2023 using the YahooProvider class from Qiskit framework [46]. Additionally, we choose a reasonable value for the budget as B = n/2.

To quantify the performance of an algorithm for solving the

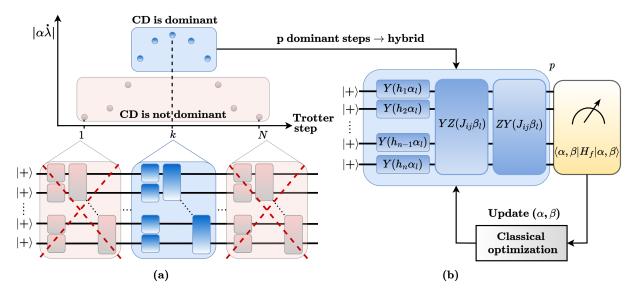


FIG. 2. (a) Purely quantum DCQO circuit. In the upper part, a qualitative behavior of  $|\alpha\lambda|$  is presented, which quantifies the dominance of the CD term. The region where it is dominant (light blue) and where it is not (light red) is shown and mapped to a quantum circuit in the bottom part. Precisely, the first, k-th, and last (N-th) step is shown. The circuit blocks coming from the first and last steps are not applied since CD is not dominant, whereas the one coming from k-th 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 ( $\alpha$ ) and another one for two-qubit rotations ( $\beta$ ). Afterward, the usual hybrid procedure is applied to optimize the cost function  $\langle \alpha, \beta | H_f | \alpha, \beta \rangle$ . 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.

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

$$r^{(\text{avg})} \equiv \frac{E_{\text{avg}} - E}{E_{\text{avg}} - E_{\text{min}}} \,. \tag{12}$$

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,  $E_{\rm avg}$  is zero and our expression reduces to  $r^{\rm (avg)} = E/E_{\rm min}$ . We can use different sets (e.g. set of feasible solutions), described by different values of  $E_{\rm avg}$ , which establishes a reference for the worst case.

## IV. IMPLEMENTATION AND RESULTS

We performed all our circuit implementations using BraKet SDK from AWS [47]. For the hybrid algorithms, we leverage The Matter Lab's Tequila framework [48] and implement an expectation value minimization routine using COBYLA as the optimization of choice and limiting to 200 iterations. Given

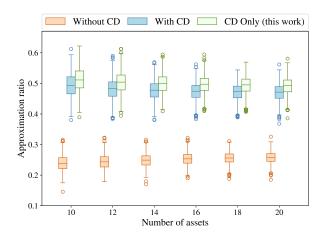


FIG. 3. Comparison of the performance of three methods: non-adiabatic (*Without CD*), inclusive of the first-order CD (*With CD*), and the 1<sup>st</sup> order CD term exclusively (*CD only*). The data represents the approximation ratio for 1,000 randomly generated Ising spin-glass instances (refer to 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.

the all-to-all connectivity of the problem at hand, we select IonQ's trapped-ion device Aria, which has 25 fully connected qubits and is commercially available through AWS's BraKet service [47]. The average one-qubit and two-qubit gate errors are 0.06% and 0.6%, respectively [47].

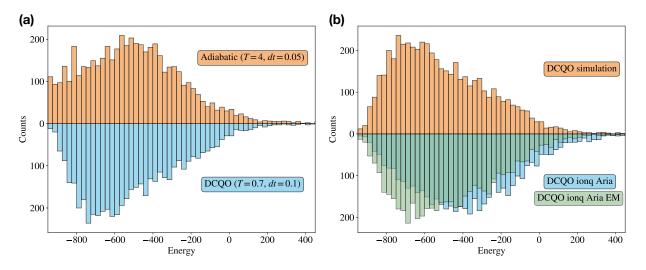


FIG. 4. (a) Energy distributions of the final state after applying DCQO (7 Trotter steps) and Adiabatic (80 Trotter steps). The approximation ratios are 0.54 and 0.53, respectively. (b) Energy distributions of the final state after applying DCQO in IonQ's 25 qubits device Aria through AWS with and without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

## 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 (12) 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  $2\times$  increase in 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.

We present simulation results for a 20-asset portfolio optimization problem in Fig. 4(a). In this figure, we observe that 7 steps of DCQO, which reduce to 1 step after gate cut-off, performs as good as 80 steps of finite-time digitized-adiabatic evolution, since  $r^{(avg)} \approx 0.54$  for both cases. This means there is a  $40\times$  reduction in the circuit depth as the depth-per-step for DCQO is twice that for finite-time digitized-adiabatic evolution.

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 A for details. Additionally, the error mitigation (EM) strategy of debiasing is added to reduce systematic errors on the hardware [49]. 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  $r^{(avg)} = 0.50$ , 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 good as a one-layer h-DCQO. Although the depth per layer for h-DCQO is twice that of QAOA, this  $5\times$  reduction in the required number of layers ultimately leads to a  $2.5\times$  reduction in circuit depth for h-DCQO as compared to QAOA. 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  $r^{(avg)} = 0.72$  for the portfolio optimization problem. The circuit-depth compression by h-DCQO is not only relevant for being able 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 purposefully 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  $r^{(avg)} = 0.58$ . 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, keeping the depth larger than DCQO circuits. Thus, further noise suppression techniques are needed in or-

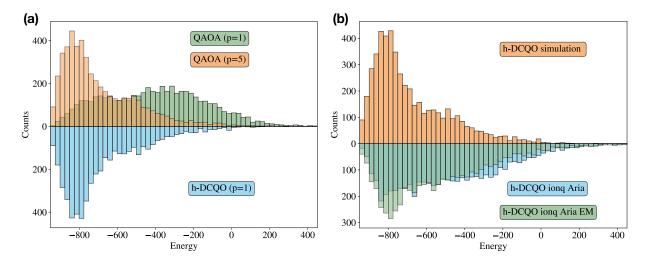


FIG. 5. (a) Energy distributions of the final state after applying h-DCQO (p = 1 layer) and QAOA (p = 1 and p = 5 layers) at the optimal parameters. The approximation ratios are 0.72, 0.41 and 0.73, respectively. (b) Energy distributions of the final state after applying h-DCQO at the optimal parameters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

der to close the gap between simulation and experiment (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. Additionally, 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 experimentally implemented our proposal on real-world portfolio data involving 20 assets and using a 20-qubit trapped-ion quantum computer. 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 toward 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: 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  $O_1 = [H_{ad}, \partial_{\lambda} H_{ad}]$  and  $O_2 = [H_{ad}, O_1]$ , i.e.  $\alpha_1 = -\Gamma_1/\Gamma_2$ , where  $\Gamma_k = ||O_k||^2$  such that the CD term takes the form

$$H_{CD} = i\dot{\lambda}\alpha_1 O_1. \tag{A1}$$

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

$$O_{1} = \left[ -\sum_{i} X_{i}, \sum_{k} h_{k} Z_{k} + \sum_{kl} J_{k < l} Z_{k} Z_{k} \right]$$

$$= 2i \sum_{i} h_{i} Y_{i} + 2i \sum_{i < j} J_{ij} \left( Y_{i} Z_{j} + Z_{i} Y_{j} \right).$$
(A2)

Therefore,  $\Gamma_1$  is given by

$$\Gamma_1 = 4 \sum_i h_i^2 + 4 \sum_{i < i} J_{ij}^2.$$
 (A3)

Now, we need to calculate the denominator of the equation for  $\alpha_1$ , which follows from  $O_2 = (1 - \lambda) [H_i, O_1] + \lambda [H_f, O_1]$ . Then, we need to evaluate two different commutators.

The first one can be written as

$$[H_i, O_1] = 4\sum_i h_i Z_i + 8\sum_{i \le j} J_{ij} Z_i Z_j - 8\sum_{i \ne j} J_{ij} Y_i Y_j,$$
(A4)

whereas the second one is

$$[H_f, O_1] = 4 \sum_{i} \left( h_i^2 + \sum_{j} J_{ij}^2 \right) X_i + 8 \sum_{i \neq j} h_i J_{ij} X_i Z_j + 8 \sum_{i < j < k} \left( J_{ij} J_{ij} X_i Z_j Z_k + J_{ij} J_{jk} Z_i X_j Z_k + J_{ik} J_{jk} Z_i Z_j X_k \right). \tag{A5}$$

It follows that,

$$\Gamma_2 = 16(1 - \lambda)^2 \left( \sum_i h_i^2 + 4 \sum_{i \neq j} J_{ij}^2 \right) + 16\lambda^2 \left( \sum_i h_i^4 + \sum_{i \neq j} J_{ij}^4 + 6 \sum_{i \neq j}^1 h_i^2 J_{ij}^2 + 6 \sum_{i < j < k} \left( J_{ij}^2 J_{ik}^2 + J_{ij}^2 J_{jk}^2 + J_{ik}^2 J_{jk}^2 \right) \right). \tag{A6}$$

Hence, the CD term is given by

$$H_{CD} = -2\dot{\lambda}\alpha_1 \left( \sum_i h_i Y_i + \sum_{i < j} J_{ij} \left( Y_i Z_j + Z_i Y_j \right) \right), \quad \text{where } \alpha_1 = -\frac{1}{4} \frac{\sum_i h_i^2 + \sum_{i < j} J_{ij}^2}{R(t)}, \tag{A7}$$

and

$$R(t) = (1 - \lambda)^2 \left( \sum_i h_i^2 + 4 \sum_{i \neq j} J_{ij}^2 \right) + \lambda^2 \left( \sum_i h_i^4 + \sum_{i \neq j} J_{ij}^4 + 6 \sum_{i \neq j}^1 h_i^2 J_{ij}^2 + 6 \sum_{i < j < k} \left( J_{ij}^2 J_{ik}^2 + J_{ij}^2 J_{jk}^2 + J_{ik}^2 J_{jk}^2 \right) \right). \tag{A8}$$

#### Appendix B: IonQ native gate implementation on Amazon Braket

The transpilation process was entirely done 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 <sup>1</sup> where they represent their native *GPi*, *GPi*2, and *MS* gate unitaries.

Logical to physical gates using single-qubit operations were translated 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 I. 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  $2\pi$  circle. We used the definition of the partially entangling MS gate as described by IonQ, this requires three parameters in all cases where the third parameter must be  $\theta \in [0, \frac{1}{4}]$  being the maximum value equal to

https://ionq.com/docs/getting-started-with-native-gates

TABLE I. Transpilation of some important gates that are present in the Hamiltonians we worked with. Specifically, we implemented in Hardware the gates H,  $R_v(\theta)$ ,  $R_{vz}(\theta)$  and  $R_{zv}(\theta)$  as shown in this table.

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

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

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

the fully-entangled MS gate implementation. Table II shows the implementation of the  $R_{yy}(\theta)$  gate depending on the third parameter restriction on the angle being selected with respect to the  $2\pi$  reference. Taking this as the basis,  $R_{yy}(\theta)$  gate 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 [49] by simply invoking it during the job submission. The main objective of this simple error mitigation technique was to not overcompensate by additional and more complex techniques.

- [1] P. W. Shor, Algorithms for quantum computation: discrete logarithms and factoring, in *Proceedings 35th annual symposium on foundations of computer science* (Ieee, 1994) pp. 124–134.
- [2] C. Gidney and M. Ekerå, How to factor 2048 bit rsa integers in 8 hours 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, T. Menke, *et al.*, Noisy intermediate-scale quantum algorithms, Reviews of Modern Physics **94**, 015004 (2022).
- [4] R. Orús, S. Mugel, and E. Lizaso, Quantum computing for finance: Overview and prospects, Reviews in Physics 4, 100028 (2019).
- [5] S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan, Quantum computational chemistry, Reviews of Modern Physics 92, 015003 (2020).
- [6] E. Farhi, J. Goldstone, and S. Gutmann, A quantum approximate optimization algorithm, arXiv preprint arXiv:1411.4028 (2014).
- [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, Physical Review 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 algo-

- rithm, arXiv preprint arXiv:2208.06909 (2022).
- [10] M. P. Harrigan, K. J. Sung, M. Neeley, K. J. Satzinger, F. Arute, K. Arya, J. Atalaya, J. C. Bardin, R. Barends, S. Boixo, *et al.*, Quantum approximate optimization of non-planar graph problems on a planar superconducting processor, Nature Physics 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, Nature communications 12, 6961 (2021).
- [12] M. Streif and M. Leib, Training the quantum approximate optimization algorithm without access to a quantum processing unit, Quantum Science and Technology 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, arXiv preprint arXiv:2307.14427 (2023).
- [15] D. Herman, C. Googin, X. Liu, Y. Sun, A. Galda, I. Safro, M. Pistoia, and Y. Alexeev, Quantum computing for finance, Nature Reviews Physics, 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 Information Processing 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, arXiv e-prints (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, Physical Review Applied 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, Physical Review Research 4, 013141 (2022).
- [23] M. Vizzuso, G. Passarelli, G. Cantele, and P. Lucignano, Convergence of digitized-counterdiabatic qaoa: circuit depth versus free parameters, arXiv preprint arXiv:2307.14079 (2023).
- [24] Q.-M. Ding, Y.-M. Huang, and X. Yuan, Molecular docking via quantum approximate optimization algorithm, arXiv preprint arXiv:2308.04098 (2023).
- [25] H. Markowitz, Portfolio selection, The Journal of Finance 7, 77 (1952).
- [26] H. Kellerer, R. Mansini, and M. G. Speranza, Selecting portfolios with fixed costs and minimum transaction lots, Annals of Operations Research 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, Cardinality constrained portfolio optimization on an ising machine, in 2022 IEEE 35th International System-on-Chip Conference (SOCC) (IEEE, 2022) pp. 1–6.
- [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, Solving the optimal trading trajectory problem using a quantum annealer, in *Proceedings of the 8th Workshop on High Performance Computational Finance* (2015) pp. 1–7.
- [31] S. Yarkoni, E. Raponi, T. Bäck, and S. Schmitt, Quantum annealing for industry applications: Introduction and review, Reports on Progress in Physics (2022).
- [32] M. Demirplak and S. A. Rice, Adiabatic population transfer with control fields, The Journal of Physical Chemistry A 107,

- 9937 (2003).
- [33] M. V. Berry, Transitionless quantum driving, Journal of Physics A: Mathematical and Theoretical 42, 365303 (2009).
- [34] A. del Campo, Shortcuts to adiabaticity by counterdiabatic driving, Physical review letters 111, 100502 (2013).
- [35] D. Sels and A. Polkovnikov, Minimizing irreversible losses in quantum systems by local counterdiabatic driving, Proceedings of the National Academy of Sciences 114, E3909 (2017).
- [36] P. W. Claeys, M. Pandey, D. Sels, and A. Polkovnikov, Floquetengineering counterdiabatic protocols in quantum many-body systems, Physical Review Letters 123, 090602 (2019).
- [37] T. Hatomura and K. Takahashi, Controlling and exploring quantum systems by algebraic expression of adiabatic gauge potential, Physical Review A 103, 012220 (2021).
- [38] K. Takahashi and A. del Campo, Shortcuts to adiabaticity in krylov space, arXiv preprint arXiv:2302.05460 (2023).
- [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, Physical Review Letters 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, Physical Review Research 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 manybody ground-state preparation inspired by counterdiabatic driving, Physical Review X 11, 031070 (2021).
- [43] Q. Xie, K. Seki, and S. Yunoki, Variational counterdiabatic driving of the hubbard model for ground-state preparation, Phys. Rev. B 106, 155153 (2022).
- [44] Getting started with native gates, https://ionq.com/docs/getting-started-with-native-gates, accessed: 2023-08-22
- [45] B. F. Schiffer, J. Tura, and J. I. Cirac, Adiabatic spectroscopy and a variational quantum adiabatic algorithm, PRX Quantum 3, 020347 (2022).
- [46] Qiskit contributors, Qiskit: An open-source framework for quantum computing (2023).
- [47] Amazon Web Services, Amazon Braket (2023).
- [48] 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 Science and Technology 6, 024009 (2021).
- [49] A. Maksymov, J. Nguyen, Y. Nam, and I. Markov, Enhancing quantum computer performance via symmetrization (2023), arXiv:2301.07233 [quant-ph].