Optimizing Parameters of Quantum Circuits with Sparsity-Inducing Coordinate Descent

Rudy Raymond , **Zichang He**
Global Technology Applied Research (GTAR), JPMorganChase & Co., New York, NY 10001, USA
{raymond.putra, zichang.he}@jpmchase.com

Abstract

Parameterized Quantum Circuit (PQC) is a family of structured quantum circuits that consists of quantum gates whose parameters are optimized with classical computers. With the quest for a potential speedup, there is a need to run larger quantum circuits, which in turn results in the arduous task of parameter optimization. In this paper, we propose a generic method, called *Rotolasso*, that utilizes sparsity-inducing coordinate descent (CD) to optimize parameters of a PQC for balancing its accuracy and the number of parameterized gates. The use of CD allows significant reduction in the number of quantum circuit runs, and the sparsity in the model leads to simpler and faster PQCs, both of which are important ingredients to overcome limitations of near-term quantum devices. We provide theoretical analyses and demonstrate experiments showing the effectiveness of *Rotolasso* to solve instances of combinatorial optimization problems.

1 Introduction

Advancement of quantum computer hardware has enabled executing quantum circuits that consists of thousands of imperfect quantum gates operating on tens to hundreds of noisy quantum bits [Arute *et al.*, 2019; Layden *et al.*, 2023]. Despite this remarkable achievement, such quantum hardware is still not sufficient to run prominent quantum algorithms, such as the prime factorization, that require several order of magnitude more quantum gates, qubits, and noise mitigation. Nevertheless, near-term quantum devices are already promising to deliver quantum advantage for non-trivial tasks in sampling [Arute *et al.*, 2019] and quantum simulation [Kim *et al.*, 2023] via their capabilities to execute quantum circuits with parameterized quantum gates.

Such quantum circuits are believed to be useful building blocks of quantum algorithms (QAs) to achieve potential quantum advantange on near-term quantum devices[Cerezo *et al.*, 2021a]. The objective of such QAs is to find the optimal parameters of quantum gates in a quantum circuit that outputs quantum states corresponding to optimal solutions by the following iterative steps: run the PQC with a parameter set initially chosen at random, measure the outcome, optimize the parameters by classical computation to obtain a better set of parameters, and go back to rerun the PQC until a stopping criterion is satisfied.

One bottleneck of the aforementioned approach of QA is the slow clock of quantum gates in PQC. There have been active research in optimizing parameters of PQCs with less quantum circuit runs [Harrow and Napp, 2021; Sweke *et al.*, 2020]. In addition, as in the classical AI/ML, the more parameters, the harder it is to optimize, and this phenomenon is more severe in PQCs as it can easily be trapped in the almost-flat landscape of objective function, a phenomenon known as *barren plateau* [McClean *et al.*, 2018; Wang *et al.*, 2021; Cerezo *et al.*, 2021b] which is akin to vanishing gradients in deep learning [Glorot and Bengio, 2010; Shalev-Shwartz *et al.*, 2017].

For these reasons, optimization strategies which are less dependent on gradient, e.g., those based on adiabatic-inspired heuristics [McClean *et al.*, 2016], parameter concentration and parameter transfer [Shaydulin *et al.*, 2023; Sureshbabu *et al.*, 2024] have been quite popular. In particular, QAOA, which utilizes $2p$ parameters of quantum circuits with p layers of non-commutative quantum operations [Farhi *et al.*, 2014], has emerged as the most promising to solve various combinatorial optimization problems. In practice, however, the value of p is severely limited by the decoherence time of near-term quantum devices, and despite its favorability over other PQCs, selecting models of QAOA with p as small as possible is crucial to improve its performance [Niu *et al.*, 2019].

Selecting and optimizing models of PQCs have been the central topics in various quantum algorithms. Like in conventional AI, many important results [Mari *et al.*, 2021; Mitarai *et al.*, 2018] with regards to PQCs are based on gradient even though at early stages there were results supporting the power of coordinate descent (CD) [Nakanishi *et al.*, 2020], or popularly introduced as *Rotosolve* in [Ostaszewski *et al.*, 2021]. With Rotosolve, one can fix all but few parameters of a PQC to solve lower-dimensional optimization problems, and iteratively perform the same optimization on other parameters until the stopping criterion is satisfied. Differs from its properties in classical machine learning, CD for PQCs seems to be a natural approach due to the *unitary condition* of quantum operation that imposes periodicity in the output of PQCs; such periodicity condition seldom holds in classical (more about this is explained in the Supplementary

Material). Nevertheless, as in classical AI, the appeal of CD for PQCs is often eclipsed by the popularity of gradient-based approaches partly due to its apparent lack of sophistication despite its acceptable performances in theory and running real quantum devices [Gujju *et al.*, 2024].

In this paper, we show the possibility of CD framework towards the goal of efficient runs of PQCs in the context of model selection. The progress of quantum hardware has significantly scaled the number of quantum operations executable on quantum devices by at least two orders of magnitude within less than a decade. While this is good for increasing the complexity of PQCs to deal with larger instances of optimization and AI tasks, optimizing their parameters become more challenging. For this reason, selecting models of PQCs balancing their number of parameters, trainability, and accuracy is crucial. There have been many proposals in this direction, but they are mostly heuristics mimicking *drop-out* by randomly omitting quantum gates and bits [Coyle *et al.*, 2024; Kobayashi *et al.*, 2022; Scala *et al.*, 2023].

Our proposed CD-based framework incorporates *model selection*, a popular statistical tool in AI [James *et al.*, 2013; von Luxburg and Schoelkopf, 2008], to efficiently execute quantum circuits used in quantum algorithms. Our method utilizes CD which is known effective for various optimization with L_1 regularization, popularly known as *Lasso* [Tibshirani, 1996]. For this reason, we call our new method *Rotolasso*. It is based on recent theoretical insights with regards to PQCs where CD/Rotosolve is known to have similar performance as its gradient-based counterparts against the barren plateau [Wada *et al.*, 2024]. Thanks to the CD framework, we can identify two sparsity opportunities to reduce quantum circuit runs; in the spectrum of functions approximating the objective value, and in the parameters of PQCs.

The sparsity in the spectrum enables us to utilize the *compressive sensing* [Candès *et al.*, 2006] to reduce the runs of quantum circuits from polynomial in the number of qubits down to proportional in the number of dominant frequencies, which is often constant in practice. The sparsity in the parameters of PQCs is similar to [Pan *et al.*, 2022]. However, thanks to the CD framework we can derive closed-form conditions to eliminate a subset of parameters, or layers/depths of QAOA circuits. The closed-form conditions can be easily extended to deal with group sparsity [Simon *et al.*, 2013], and can be combined with a recent heuristic proposal to optimize parameters of QAOA with *Fourier* [Zhou *et al.*, 2020]. We show the efficacy of CD-based model selection by experimenting on instances of MaxCut and LABS, two hard problems with promising near-term quantum algorithms. Our proposal shed lights on AI techniques, such as model selection, that are useful to improve quantum computing. Supplementary Material is available at 10.5281/zenodo.15425252.

2 Related Work

Because parameter optimization of PQC is NP hard, one has to rely on heuristic optimizers classified into those based on gradient, gradient-free, and machine learning. The gradient-based optimizers were first introduced in [Guerreschi and Smelyanskiy, 2017; Wang *et al.*, 2018] while Stochastic Gradient Descent (SGD), was shown also effective for finite number of measurements of QAOA runs in [Sweke *et al.*, 2020]. Unlike the classical case, evaluating gradients are difficult and costly, thus, gradient-free optimizers, such as, Nelder-Mead or SPSA are often preferred in practice. [Hao *et al.*, 2024] proposed an end-to-end protocol for setting high quality QAOA parameters efficiently. A review on setting QAOA parameters is presented in [He *et al.*, 2024].

Our proposed model selection falls into heuristics based on combining gradient-free and machine-learning based optimizers. It is based on the gradient-free *Rotosolve* first introduced in [Nakanishi *et al.*, 2020; Ostaszewski *et al.*, 2021] with the first proposal of model selection using L_1 regularization in [Pan *et al.*, 2022]. Meanwhile, other machine-learning based optimizers include those based on Bayesian optimization [Otterbach *et al.*, 2017], Kernel Density Estimation (KDE) [Khairy *et al.*, 2020], and Reinforcement Learning (RL) [Yao *et al.*, 2020; Wauters *et al.*, 2020].

3 Preliminaries

We start by describing combinatorial optimization in the form of Binary Optimization and PQCs with emphasis on QAOA. For ease of explanation, the binary optimization is on variables whose values are either -1 or 1 (instead of 0 or 1); switching between those pairs of values is trivial.

3.1 Combinatorial Optimization

We consider (combinatorial) binary optimization problem in the following unconstrained minimization problem:

$$\min_{\boldsymbol{z}\in\{-1,1\}^n} f(\boldsymbol{z}), \tag{1}$$

where $\boldsymbol{z}$ is an n-dimensional binary vectors representing decision variables, and $f : \mathbb{R}^n \to \mathbb{R}$ is the objective function.

MaxCut. A typical example of binary optimization is finding optimal bipartition (or, cut) of graph $G(V, E)$, where $V = \{1, \ldots, n\}$ is the node set of G, and $E = \{(i, j) \in [n]^2\}$ is its edge set. For simplicity, in this paper we only discuss unweighted graphs. A cut value of the graph is the sum of edges connecting vertices with different labels. *MaxCut* is a problem to label each node i of G with $z_i \in \{-1, 1\}$ to maximize the cut value $\max_{\boldsymbol{z}\in\{-1,1\}^n} \frac{1}{2}\sum_{(i,j)\in E}(1 - z_i z_j)$. In the standard form it can be formulated as

$$\min_{\boldsymbol{z}\in\{-1,1\}^n} \frac{1}{2}\sum_{(i,j)\in E} z_i z_j, \tag{2}$$

because $\sum_{(i,j)\in E} 1/2 = |E|/2$ is constant. Due to its NP-hardness, almost all optimization algorithms solve the relaxation by treating $\boldsymbol{z} \in \mathbb{R}^n$. The quality of an algorithm for MaxCut is measured in *Approximation Ratio* (AR) which is defined as the ratio of its cut value against the optimal one.

MaxCut on $G(V, E)$ can be solved on quantum computers by constructing its qubit *Hamiltonian*, which is essentially an $2^n \times 2^n$ complex matrix so that the eigenvector (or, *ground state*) corresponding to its smallest eigenvalue (or, *ground state energy*) encodes the optimal solution. The Hamiltonian of MaxCut is given as

$$H_{MC} = \frac{1}{2}\sum_{(i,j)\in E} Z_i Z_j, \tag{3}$$

where $Z_i \equiv \underbrace{I_2 \otimes \ldots \otimes I_2}_{i-1} \otimes Z \otimes \underbrace{I_2 \otimes \ldots \otimes I_2}_{n-i}$, such that I_2 is the 2×2 identity matrix, and Z is the 2×2 Pauli Z matrix which is a diagonal matrix. Notice the relation between Eq. (3) and Eq. (2): z_i is replaced by Z_i, and vice versa. We can also confirm that H_{MC} is a diagonal $2^n \times 2^n$ matrix whose (i,i)-th element is the value of Eq. (2) corresponding to the binary representation of i.

LABS. Another hard problem we consider is the so-called *low-autocorrelation binary sequences* (LABS) [Shaydulin *et al.*, 2024] whose objective is to minimize the sum of the squares of autocorrelations of $\boldsymbol{z} \in \{-1,1\}^n$ formulated as

$$\min_{\boldsymbol{z} \in \{-1,1\}^n} \sum_{k=1}^{n-1} \left(\sum_{i=1}^{n-k} z_i z_{i+k} \right)^2. \tag{4}$$

It has applications in communication engineering and statistical mechanics [Golay, 1977; Bernasconi, J., 1987]. Some aspects of LABS are its optimal solutions are known only for $n \leq 66$ [Packebusch and Mertens, 2016], and its instance is unique for each n. The quality of an algorithm for LABS outputting $\boldsymbol{z}$ is measured in *Merit Factor* (MF) which is defined as $\mathrm{MF}(\boldsymbol{z}) = n^2 / (2E(\boldsymbol{z}))$, where $E(\boldsymbol{z})$ is the value of the function inside the min in Eq. (4). Similar to MaxCut, LABS can be solved on quantum computers by constructing its qubit Hamiltonian: by substituting Z_i for z_i in Eq. (4) we can obtain an $2^n \times 2^n$ matrix whose ground state corresponds to the optimal solution of Eq. (4).

3.2 Parameterized Quantum Circuits (PQCs)

Here, we give basic concepts of PQCs necessary while more are given in the Supplementary Material. For details of quantum computing and PQCs, please see, e.g., [Nielsen and Chuang, 2010; Cerezo *et al.*, 2021a].

PQCs are promising tools to utilize near-term quantum devices for combinatorial optimization tasks. They include variational quantum circuits (VQCs) whose typical circuit is depicted in Fig. 1 (a), and Quantum Approximate Optimization Algorithm (QAOA), whose typical circuit is shown in Fig. 1 (b). Notice that both circuits are executed from the same initial quantum states, and both consist of repeated layers of entangling gates. There are two main differences between VQC and QAOA which can be observed from the figures. First, the structure of the entangling gates in VQC is independent of the minimization problem, while that in QAOA depends on the minimization problem through the "Cost" block. For example, in Fig. 1 (b), the two-qubit gates are applied on the i-th and j-th qubits if and only if $z_i z_j$ appears in the objective function (e.g., Eq. (2)). Secondly, the parameters of the quantum gates in VQC can take different values, while those in QAOA are the same if they belong to the same block.

Despite theoretical evidences that VQC may suffer from the barren-plateau problems [Cerezo *et al.*, 2021b], experimental evidences showed possibility of quantum advantage for QAOA on LABS [Shaydulin *et al.*, 2024] and other optimization tasks [Bravyi *et al.*, 2022]. On the other hand, many experimental evidences point to the potential of VQC for MaxCut and could find better solutions than QAOA with less circuit depth or layers, e.g., in [Kondo *et al.*, 2024]. Our proposed solution is demonstrated on both VQC and QAOA.

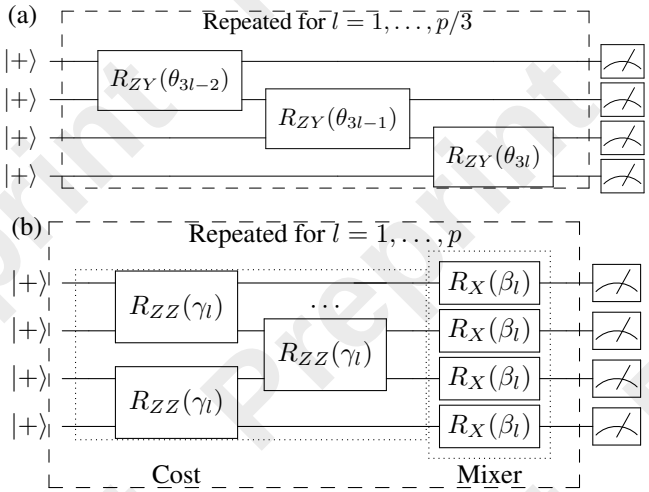

Figure 1: (a) An example of 4-qubit VQC with $R_{ZY}(\theta_i)$ two-qubit (or, entangling) gates. Each line in the circuit denotes a qubit, whose initial quantum state is $|+\rangle \equiv 1/\sqrt{2}(|0\rangle + |1\rangle)$. Notice that the structure of the circuit (i.e., alignment of quantum gates or ansatz), is independent of the objective function. (b) An example of 4-qubit QAOA circuit with a block of R_{ZZ} gates with parameter γ_l (marked as "Cost") and a block of R_X gates with parameter β_l (marked as "Mixer") at each layer l for realizing $U(\beta_l, \gamma_l)$ as in Eq. (5). The block of R_{ZZ} gates is for realizing $e^{-i\gamma_l H_C}$, which depends on the objective function to be optimized, while that of R_X gates is for $e^{-i\beta_l H_M}$ and does not depend on the objective function.

There are two main steps for the utility of PQCs for combinatorial optimization: constructing a Hamiltonian H of the optimization problem and approximating the ground state $|\psi\rangle \in \mathbb{C}^{2^n}$ of H from the optimal parameters of PQCs. The solution of the combinatorial optimization can be read from the ground state. The Hamiltonians corresponding to MaxCut and LABS were introduced in the previous section.

The second step of using PQCs is to approximate the ground state of the Hamiltonian. The quantum state generated by a VQC, whose example shown in Fig. 1 (a), is parameterized by $\boldsymbol{\theta} \equiv (\theta_1, \ldots, \theta_m)$, where each $\theta_i \in \mathbb{R}$ is the parameter of the i-th single-qubit gate of the PQC. Similarly, as shown in the instance of QAOA circuit in Fig. 1 (b), a p-layer QAOA is parametrized by $\boldsymbol{\theta} \equiv (\gamma_1, \beta_1, \ldots, \gamma_p, \beta_p)$, where $\gamma_l \in \mathbb{R}$ and $\beta_l \in \mathbb{R}$ are the parameters for the l-th sublayer of the *Cost* Hamiltonian H_C and the *Mixing* Hamiltonian H_M, respectively. Here, the cost and mixing Hamiltonian of an n-qubit QAOA system for the MaxCut are, respectively,

$$H_M = \sum_{i=1}^{n} X_i, \quad H_C = \sum_{(i,j) \in E} Z_i Z_j,$$

where X_i is defined similarly as the Z_i, albeit Z at the i-th qubit is replaced by the Pauli X matrix. Meanwhile, for LABS, the mixing Hamiltonian is the same but the corresponding H_C becomes its qubit Hamiltonian.

The l-th layer of the n-qubit system QAOA is defined by

the following unitary transformation

$$U(\beta_l, \gamma_l) = e^{-i\beta_l H_M} e^{-i\gamma_l H_C}, \tag{5}$$

so that the (pure) quantum state after the p-th layer becomes

$$|\psi_p\rangle = U(\beta_p, \gamma_p) \dots U(\beta_1, \gamma_1) |\psi_0\rangle, \tag{6}$$

where the initial quantum state $|\psi_0\rangle$ is $|+\rangle^{\otimes n}$, which is the ground state of the mixing Hamiltonian H_M. QAOA is inspired by adiabatic process of evolving the ground state of Hamiltonian $H = \alpha H_C + (1-\alpha) H_M$ from $\alpha = 0$ to $\alpha = 1$.

The optimization of QAOA's parameters is formulated as

$$\{(\beta_l^*, \gamma_l^*)\}_{l=1}^p = \arg \min_{\{(\beta_l, \gamma_l)\}_{l=1}^p} \langle \psi_p | H_C | \psi_p \rangle, \tag{7}$$

where $\langle \psi_p|$ is the conjugate transpose of $|\psi_p\rangle$.

The challenge in utilizing QAOA is to find the optimal $\{(\beta_l^*, \gamma_l^*)\}_{l=1}^p$, which can be extremely difficult for large p. There are various proposal to optimize them by gradient-based methods [Wierichs *et al.*, 2022], or by gradient-free methods such as *Cobyla*, or by extracting generic and universal patterns that can be transferred to other instances as in *Fourier* [Zhou *et al.*, 2020]. Nevertheless, they still require huge computational resources and communication between quantum devices and classical co-processors so that research on QAOA with as small p is possible is still active. Additionally, [Choi *et al.*, 2023] shows the smaller parameter θ_i of QAOA, the easier it is to detect and correct noise of near-term quantum devices. These strongly suggest the importance of restricting parameters of QAOA by regularization.

3.3 Sparsity-Inducing L_1-regularization

Learning over-parameterized models has been very successful and has been the backbone to the success of neural network and its applications. Suppose that $f(\boldsymbol{\theta}) \in \mathbb{R}$ is the function with parameters $\boldsymbol{\theta} \in \mathbb{R}^{2p}$ to be optimized. *Lasso* [Tibshirani, 1996] is a popular method to optimize $f(\cdot)$ so that most of the elements of $\boldsymbol{\theta}$ are zeroes (or, sparse). The sparsity is considered effective for improving the efficacy and interpretability of the statistic model associated with $f(\cdot)$. The objective function for inducing *sparsity* in $\boldsymbol{\theta}$ is

$$\min_{\boldsymbol{\theta}} f(\boldsymbol{\theta}) + \lambda \|\boldsymbol{\theta}\|, \tag{8}$$

where $\lambda > 0$, and $\|\boldsymbol{\theta}\|$ is the L_1-norm defined as $\|\boldsymbol{\theta}\| = \sum_{i=1}^{2p} |\theta_i|$. CD and its variants are popular methods for solving this optimization problem [Wright, 2015].

4 Sparsity-Inducing Coordinate Descent

We propose a CD method to induce sparsity in the parameters of PQCs in general, and of QAOA in particular. For QAOA, the resulting parameters are expected to be those with small p and smaller θ_i as the sparsity in $\boldsymbol{\theta}$ means that most of (β_i, γ_i) are zeroes and hence can be omitted.

To unifiedly explain our proposed solution, based on Eq. (7), let us denote $f(\boldsymbol{\theta})$ as

$$f(\boldsymbol{\theta}) \equiv \langle \psi_p | H_C | \psi_p \rangle,$$

where for simplicity, we write $\boldsymbol{\theta} = (\theta_1, \dots, \theta_{2p}) = (\gamma_1, \beta_1, \dots, \gamma_p, \beta_p)$, i.e., $\theta_{2i-1} = \gamma_i$ and $\theta_{2i} = \beta_i$ for $i = 1, \dots, p$, and $|\psi_p\rangle$ is the quantum state output by the PQC with parameters $\boldsymbol{\theta}$.

Our proposed solution starts from the following observation, first shown in [Nakanishi *et al.*, 2020] that if we fix all θ_i except at $i = j$, then $f(\cdot)$ is a univariate function that can be written as $\hat{f}(\cdot)$ according to

$$\hat{f}(\theta_j) = c_{0,j} + \sum_{k=1}^{m} c_{k,j} \cos(k\theta_j) + \sum_{k=1}^{m} c_{m+k,j} \sin(k\theta_j), \tag{9}$$

where m is the number of gates with the same parameter θ_j. That is, for VQC it is usually $m = 1$, while for QAOA it is $m = n$ when j is even and m equals to the number of zz terms when j is odd. Eq. (9) provides an opportunity to apply CD without regularization. Namely, by keeping $\boldsymbol{\theta}^{(t)} = (\theta_1^{(t)}, \dots, \theta_{2p}^{(t)})$ at each iteration $t = 1, \dots, T$ one can update the value of $\theta_j^{(t)}$, for $j \equiv t \mod (2p)$, by the rotation angle that (globally) minimizes Eq. (9). For completeness, the procedure is given as below, see, e.g., [Wada *et al.*, 2024].

Coordinate Descent (CD)

- Evaluate VQC or QAOA at $2m + 1$ different rotation values to obtain the tuples $\{(\theta_{j,l}, \hat{f}(\theta_{j,l}))\}_{l=1}^{2m+1}$.
- Build a system of linear equations as Eq. (9) for obtaining $(c_{0,j}, c_{1,j}, \dots, c_{2m,j})$ with classical computers from $\{(\theta_{j,l}, \hat{f}(\theta_{j,l}))\}_{l=1}^{2m+1}$.
- Compute $\theta_j^* = \arg\min_{\theta_j} \hat{f}(\theta_j)$ by classical computers. This can be computed efficiently from the solution of the previous step and grid search.

When optimizing PQCs with *state-vector* simulators, the number of quantum circuit executions is $O(m)$, while the rest of the computational cost is poly(m) of classical computation. For VQC, m can be made constant, but for QAOA it can grow quadratically in the number of qubits n which can be too costly for dealing with large instances of optimization problems. Fortunately, we can significantly reduce it to $O(\kappa \log m)$, where κ is the number of non-zero coefficients of Eq. (9), by applying compressive sensing [Candès *et al.*, 2006] to approximate Eq. (9).

4.1 Sparse Approximation of Univariate Functions

For QAOA circuits m can be large and simulating them with quantum simulators on classical computers can be very time-consuming. The compressive sensing can be utilized to reduce quantum cirtcuit runs by finding a function $\hat{g}(\theta_j)$ that approximates $\hat{f}(\theta_j)$ in the form of

$$\hat{g}(\theta_j) = c'_{0,j} + \sum_{k=1}^{m} c'_{k,j} \cos(k\theta_j) + \sum_{k=1}^{m} c'_{m+k,j} \sin(k\theta_j), \tag{10}$$

so that $(c'_{0,j}, c'_{1,j}, \dots, c'_{2m,j})$ is sparse and $\hat{g}(\theta_j) = \hat{f}(\theta_j)$ for $j = 1, \dots, \tilde{m}$ for $\tilde{m} \ll m$. The sparse approximation function $\hat{g}$ can be computed by slightly modifying the first two steps of CD as below.

- Evaluate QAOA circuits at $\tilde{m} \ll m$ random values of $\{\theta_{j,l}\}_{l=1}^{\tilde{m}}$ to obtain $\{\hat{f}(\theta_{j,l})\}_{l=1}^{\tilde{m}}$.

- Let A be the $\tilde{m} \times (2m+1)$ matrix whose i-th row is $(1, \cos(\theta_{j,i}), \ldots, \cos(m\theta_{j,i}), \sin(\theta_{j,i}), \ldots, \sin(m\theta_{j,i}))$. Let $\boldsymbol{f}$ be a column vector $(\hat{f}(\theta_1), \ldots, \hat{f}(\theta_{\tilde{m}}))^T$, and $\boldsymbol{c}$ be a $(2m+1)$-dimensional column vector. Then, the sparse $\hat{g}(\cdot)$ approximating $\hat{f}(\cdot)$ can be computed from the following linear programming (LP) so that the values of $(c'_{0,j}, c'_{1,j}, \ldots, c'_{2m,j})$ can be read from $\boldsymbol{c}^*$.

$$\begin{aligned} \boldsymbol{c}^* = \arg\min_{\boldsymbol{c}} \quad & \|\boldsymbol{c}\| \\ \text{s.t.} \quad & A\boldsymbol{c} = \boldsymbol{f} \end{aligned} \tag{11}$$

According to [Candès *et al.*, 2006], $\tilde{m} = O(\kappa \log m)$ suffices which means exponentially fewer runs of quantum circuits than CD in the previous section. In practice it is reported that even $\tilde{m} = O(\kappa)$ is sufficient. Once we have $\hat{g}(\cdot)$, we can use it to compute the optimal rotation parameter and to guide finding the sparse model of PQCs, which constitutes our most important contribution. For simplicity, in the hereafter we use $\hat{g}(\cdot)$ and $\hat{f}(\cdot)$ interchangeably.

When optimizing PQCs on real quantum devices (or, non-state-vector simulators), we can only obtain the approximated values of $f(\cdot)$ by repetition of quantum circuit runs that has to be measured with many shots. Fortunately, the number of shots to obtain samples of $f(\cdot)$, which are sufficient to solve the Eqs. (9)–(11), is only polynomial in the size of the optimization instances, and can be further reduced with the compressive sensing. We state them in the following theorem and corollary whose proofs are in the Supplementary Material.

Theorem 1. *Let $\mathbf{c}$ is the optimal solution of the $(2m+1)$ linear system of equalities as in Eq.* (10) *under the noiseless data $\boldsymbol{f}$ satisfying $A\mathbf{c} = \boldsymbol{f}$, and let $\tilde{\mathbf{c}}$ be that under noisy circuit samplings $\tilde{\boldsymbol{f}}$ satisfying $A\tilde{\mathbf{c}} = \tilde{\boldsymbol{f}}$. For $0 < \delta, \epsilon \ll 1$. Then, there exist algorithms that with probability at least $1-\delta$ output $\tilde{\mathbf{c}}$ which is ϵ-near to $\mathbf{c}$, i.e., $\|\tilde{\mathbf{c}} - \mathbf{c}\| \leq \epsilon$, by sampling the quantum circuits for $O\left(\frac{m^2|E|^2 \ln(m/\delta)}{\epsilon^2}\right)$ times on MaxCut instance of $G(V,E)$, and for $O\left(\frac{m^2 n^6 \ln(m/\delta)}{\epsilon^2}\right)$ times on LABS instance of n variables.*

Corollary 1. *Let $\mathbf{c}$ and $\tilde{\mathbf{c}}$ be the noiseless and noisy vectors defined similarly for the solution of the compressive sensing for $\tilde{m} \ll m$ as in Eq.* (10)*. Then, there exist algorithms that with probability at least $1-\delta$ output $\tilde{\mathbf{c}}$ which is ϵ-near to $\mathbf{c}$ by sampling the quantum circuits for $O\left(\frac{\tilde{m}^2|E|^2 \ln(\tilde{m}/\delta)}{\epsilon^2}\right)$ times on MaxCut instance of $G(V,E)$, and for $O\left(\frac{\tilde{m}^2 n^6 \ln(\tilde{m}/\delta)}{\epsilon^2}\right)$ times on LABS instance of n variables.*

4.2 Sparse PQC Models

Now, we can conveniently represent a subproblem of the p layers of QAOA or any other parameterized quantum circuits in Eq. (8) from the univariate trigonometric polynomials in Eq. (9) and Eq. (10). Our proposal of using sparsity-inducing CD aims at solving Eq. (8). Namely, for $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_{2p})$ we will find optimal schedules in which zero θ_i's are preferred. Let us fix all θ_i except at j to have Eq. (9) for simplicity. We can turn the trigonometric polynomial into algebraic

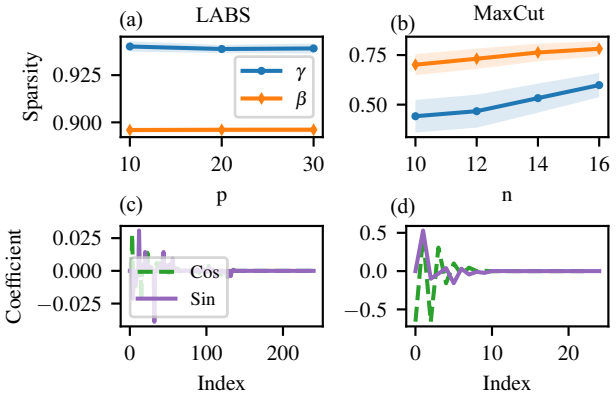

Figure 2: Sparsity of the coefficients of Eq. (9) for QAOA on MaxCut and LABS instances. (a) plots the average sparsity over p parameters of γ and β for $n = 10$ LABS problem with $p = 10, 20, 30$. Each data point is derived from 10 random initial $\boldsymbol{\theta}$'s. (c) shows the coefficients of cos and sin basis at $p = 30$ for one γ parameter under one initialization. (b) plots the average sparsity over p parameters of γ and β when $p = 10$ on graphs with $n = 10, 12, 14, 16$ nodes. Each data point is derived from 5 random initial $\boldsymbol{\theta}$'s. (d) shows the coefficients of cos and sin basis at $n = 16$ for one γ parameter under one initialization. The shaded regions represent the standard errors.

polynomial $h(x_j)$ by variable transformation as

$$x_j = \tan\left(\frac{\theta_j}{2}\right),\ \cos(\theta_j) = \frac{1-x_j^2}{1+x_j^2},\ \sin(\theta_j) = \frac{2x_j}{1+x_j^2},$$

which transforms $f(\boldsymbol{\theta})$ in Eq. (8) into a multivariate algebraic polynomial $h(\boldsymbol{x})$, and Eq. (8) becomes as in the following optimization problem

$$\min_{\boldsymbol{x} \in \mathbb{R}^{2p}} h(\boldsymbol{x}) + \lambda \|\boldsymbol{x}\|, \tag{12}$$

for $\frac{\partial h(\boldsymbol{x})}{\partial x_j} = \frac{\mathrm{d}\hat{h}(x_j)}{\mathrm{d}x_j}$ where

$$\hat{h}(x_j) = \frac{1}{(1+x_j^2)^n}\left(d_0 + \sum_{k=1}^{2n} d_k x_j^k\right), \tag{13}$$

and $d_k \in \mathbb{R}$. From this, we can reparametrize $\boldsymbol{\theta}$ with $\boldsymbol{x}$ and obtain sparse $\boldsymbol{\theta}$ from sparse $\boldsymbol{x}$.

From the above equations, we can confirm

$$\frac{\mathrm{d}\hat{h}(x_j)}{\mathrm{d}x_j} = \frac{\left(\sum_{k=1}^{2n} k d_k x_j^{k-1}\right)\left(1+x_j^2\right) - 2n x_j \left(d_0 + \sum_{k=1}^{2n} d_k x_j^k\right)}{\left(1+x_j^2\right)^{n+1}},$$

while

$$\frac{\partial \lambda|\boldsymbol{x}|}{\partial x_j} = \begin{cases} -\lambda, & \text{if } x_j < 0, \\ [-\lambda, \lambda], & \text{if } x_j = 0, \\ \lambda, & \text{if } x_j > 0. \end{cases}$$

By the stationary condition, from the above equations we observe that for $x_j = 0$ to be a local optima, the necessary condition is

$$\lambda \geq \left|\frac{\partial h(\boldsymbol{x})}{\partial x_j}\Big|_{x_j=0}\right| = \left|\frac{\mathrm{d}\hat{h}(x_j)}{\mathrm{d}x_j}\Big|_{x_j=0}\right| = |d_1|,$$

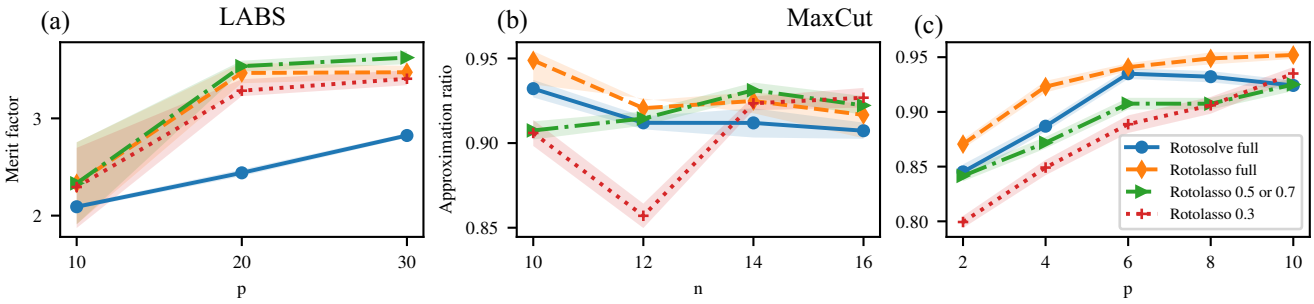

Figure 3: Performance of QAOA on MaxCut and LABS instances. (a) For LABS instance $n = 10$, we show the results under different $p = 10, 20, 30$ over different models: Rotosolve, *Rotolasso* with full (exact) linear equations, and *Rotolasso* with compressive sensing keeping 50% and 30% of the γ's while 50% of the β's. Each point is derived from 5 random initial $\boldsymbol{\theta}$'s. (b) Fixing $p = 8$ and varying $n = 10, 12, 14, 16$ MaxCut instances, we compare Rotosolve, *Rotolasso* with full linear equations, and *Rotolasso* with compressive sensing keeping 70% and 30% of the γ's while 50% of the β's. Each point is derived from 5 initial $\boldsymbol{\theta}$'s and 5 random graphs. (c) Fixing $n = 10$ MaxCut instances and varying $p = 2, 4, 6, 8, 10$, we perform similar comparison as in (c). Each point is derived from 5 random initial $\boldsymbol{\theta}$'s and 5 random graphs. The shaded regions represent the standard errors.

where $d_1 = 2\sum_{k=1}^{m} k c_{m+k,j}$ because $\frac{\mathrm{d}\hat{h}}{\mathrm{d}x_j} = \frac{\mathrm{d}\hat{g}(\theta_j)}{\mathrm{d}\theta_j}\frac{\mathrm{d}\theta_j}{\mathrm{d}x_j}$ by the chain rule. Thus, having $\hat{f}(\cdot)$, i.e., $(c_{0,j}, c_{1,j}, \ldots, c_{2m,j})$, we can obtain both the global optimal of $\hat{f}(\cdot)$, and the condition of θ_j to become zero. The update rule of $\boldsymbol{\theta}^{(t)}$ for sparsity-inducing CD is as the following: at the t-th iteration for $t = 1, 2, \ldots, T$, the value of $\theta_j^{(t)} = \theta_j^{(t-1)}$ if $j \neq t \mod 2p$, and otherwise if $j \equiv t \mod 2p$,

$$\theta_j^{(t)} = \begin{cases} 0 & , \text{ if } |d_1| \leq \lambda, \\ \arg\min_{\theta_j} \hat{f}(\theta_j) & , \text{ otherwise.} \end{cases} \tag{14}$$

5 Numerical Results

Experiments demonstrating the role of sparsity-inducing CD in PQCs are conducted using *qujax* [Duffield *et al.*, 2023] for classical simulation of quantum circuits of both QAOA and VQC for LABS and MaxCut (3-regular graphs) instances. We compare our method with Rotosolve that has been shown to be significantly better than state-of-the-art methods, such as SPSA, BFGS, CG and others [Nakanishi *et al.*, 2020].

Following the standard usage of L_1-regularized optimizer in conventional machine learning, we iteratively run *Rotolasso* on a randomly initialized $\boldsymbol{\theta}^{(0)}$ with the regularization factor λ initialized to λ_{start}, and update each parameter cyclically to obtain a (sub)optimal $\boldsymbol{\theta}^*$. The value of λ is then reduced by a factor while utilizing the previously obtained $\boldsymbol{\theta}^*$ as the new initial parameter set. This is repeated until λ becomes very close to 0. In our experiments, λ_{start} is additionally divided by the number of parameters.

5.1 Sparsity of Univariate Functions

The sparsity of coefficients in Eq. (9) is particularly important for justifying Eq. (10), the use of compressive sensing, and hence reducing quantum circuit runs for QAOA. Fig. 2 (a) and (b) depict the sparsity, i.e., the ratio of number of coefficients whose absolute values are smaller than 1e-4 over the total absolute sum of coefficients. We can see that the sparsity is high for a LABS instance; the ratios of zero coefficients are between 90%–94% and they are less p dependent. As seen from the subfigure (c), the magnitude of the coefficients are small (there are roughly 50 dominant coefficients out of 250 ones). Meanwhile, the sparsity is less outstanding at instances of MaxCut with the ratios of zero coefficients are between 35% – 80%. That is because the QAOA circuit of LABS is much more complicated than the MaxCut one. The sparsity is also independent of n, implying that we can be more aggressive in compressive sensing to deal with large-size instances. Subfigure (d) shows one example of the coefficients for a MaxCut instance.

5.2 Rotolasso on QAOA

The effectiveness of regularized QAOA parameters is demonstrated by the experiments, as shown in Fig.3. In subfigure (a), we illustrate that *Rotolasso*, both with and without compressive sensing, achieves better MF than Rotosolve on LABS instances. The improvement over Rotosolve becomes more pronounced as p increases. Additionally, we show that the sparsity within Eq.(9) can be leveraged, potentially leading to even better results using less circuit runs.

Regarding MaxCut instances, subfigure (b) shows that *Rotolasso*, with and without compressive sensing, outperforms Rotosolve across different instance sizes n. Subfigure (c) demonstrates that *Rotolasso* consistently performs better than Rotosolve for all values of p. Additionally, as p increases, the AR of *Rotolasso* with compressive sensing improves.

Fig. 4 (a) and (c) show the sensitivity of *Rotolasso* to λ_{start}. The performance ratio, defined as the ratio of *Rotolaso*'s AR to Rotosolve's AR, is close to 1 when λ_{start} is very small (e.g., 10^{-7}). However, across a wide range of λ_{start}, the AR of *Rotolasso* is generally slightly better than that of Rotosolve, indicating the robustness of *Rotolasso* with respect to the regularization term. Although the performance ratios are generally close to 1, indicating that Rotosolve performs well for QAOA in these MaxCut instances, the final $|\boldsymbol{\theta}|_1$ under a large λ_{start} is significantly smaller than that of Rotosolve. This makes it less prone to errors when executed on a quan-

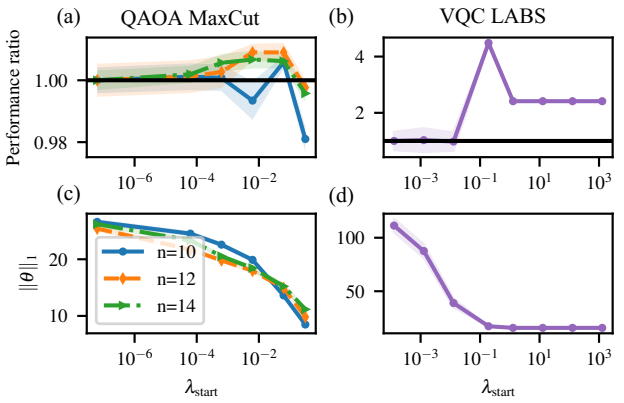

Figure 4: The sensitivity of the initial value of the regularization λ_{start} term of *Rotolasso*. In the first column, we analyze QAOA for MaxCut instances of graphs with $n = 10, 12, 14$ fixing $p = 8$ using *Rotolasso* without compressive sensing. (a) shows the final ratio between the AR of *Rotolasso* against that of Rotosolve as λ_{start} increases. (c) shows the final $\|\boldsymbol{\theta}\|_1$ as λ_{start} increases. Each datapoint is derived from 50 random initial $\boldsymbol{\theta}$. In the second column, we analyze *Rotolasso* on VQC for a LABS instance with $n = 13$. Each data point is derived from 10 random initial $\boldsymbol{\theta}$'s. (b) shows the final ratio between MF of *Rotolasso* against Rotosolve for LABS instances as λ_{start} increases. (d) shows the final $\|\boldsymbol{\theta}\|_1$ of the VQC optimized with *Rotolasso* whose regularization term is initialized with λ_{start}.

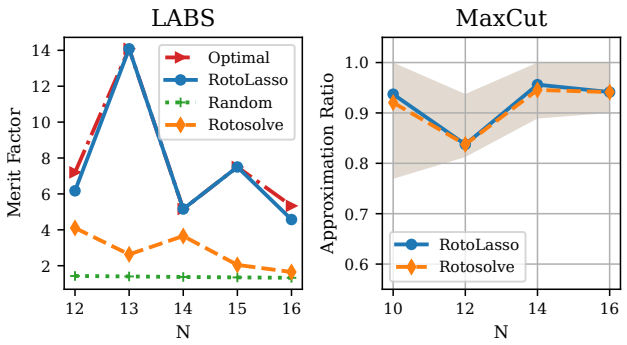

Figure 5: The effectiveness of *Rotolasso* measured in Merit Factor (MF) against Rotosolve on instances of LABS for $12 \leq n \leq 16$. *Optimal* and *Random* are, resp., the provably optimal MF and the average MF of random assignment.

tum processor, where errors are known to be proportional to absolute value of the rotation angles [Moses *et al.*, 2023].

5.3 Rotolasso on VQC

We use a n-qubit VQC with one layer of R_{ZY} gates on every unique pair of qubits as the ansatz to solve the LABS problem, which is the same as in [Morris and Lotshaw, 2024].

The results on instances of LABS for $12 \leq n \leq 16$ are as shown in Fig. 5. It can be observed that *Rotolasso* is significantly better than Rotosolve and as a matter of fact, the optimal MF was obtained for $n = 13, 14, 15$. Interestingly, *Rotolasso* is also better for this range of n when compared to the prior benchmark using QAOA [Shaydulin *et al.*, 2024].

We can also see how the regularization strength λ affects the quality of the objective function obtained by the proposed solution in Figs. 4. We can confirm that the L_1 norm of the optimized parameters indeed tends to decrease as the initial value of the regularization strength increases as depicted in Fig. 4 (d). Moreover, the Merit Factor (MF) is low when λ starts from λ_{start} which is too small, such as 10^{-4}. This implies Rotosolve without regularization easily trapped into bad local optimal. On the other hand, when λ_{start} is too big, such as 10^3, the MF is not optimal but still much better than the too-small λ_{start} as shown in Fig. 4 (b) where the performance ratio is defined as the ratio of *Rotolasso*'s against Rotosolve's, implying that the L_1 regularization indeed helps improving the solution. The figure also shows that there is a range of Goldilocks zone of λ_{start} that provides better MFs, which is around $10^{-1} - 10^{0}$ in Fig. 4 (b). We also observe that under a large range of λ_{start}, the VQC parameter optimization results are less dependent on the initialization, which is also different from the QAOA results.

6 Concluding Remarks

In this study, we propose a method to optimize parameters of quantum circuits pertaining to solving optimization problems by combining gradient-free coordinate descent with sparsity-inducing L_1 regularization. The method was designed to benefit from the sparsity of functions representing the objective values obtained from the output state of quantum circuits so that compressive sensing can be utilized to further reduce execution of quantum circuits. We derive analytical rules of updating the parameters based on the closed-form solution of coordinate-descent method. Through numerical experimentation, the effectiveness of the proposed method was demonstrated, especially on the efficacy and robustness of the regularization strength to improve the quality of the solutions of parameterized quantum circuits. The proposed method can be run on top of the recently proposed *random* CD whose total number of iterations is shown theoretically the same as that of the full gradient descent [Ding *et al.*, 2024]. Our new method thus sheds light on the application of model selection techniques in machine learning to assist in quantum computing.

Acknowledgments

The authors would like to thank Shree Hari Sureshbabu for his helps on experiments and proof readings, Dylan Herman for his theoretical advise, Marco Pistoia for his supervision of the project, and the Quantum Platform team at GTAR for technical supports.

References

[Arute *et al.*, 2019] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, October 2019.

[Bernasconi, J., 1987] Bernasconi, J. Low autocorrelation binary sequences : statistical mechanics and configuration space analysis. *J. Phys. France*, 48(4):559–567, 1987.

[Bravyi *et al.*, 2022] Sergey Bravyi, Alexander Kliesch, Robert Koenig, and Eugene Tang. Hybrid quantum-classical algorithms for approximate graph coloring. *Quantum*, 6:678, March 2022.

[Candès *et al.*, 2006] Emmanuel J. Candès, Justin K. Romberg, and Terence Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59(8):1207–1223, 2006.

[Cerezo *et al.*, 2021a] M. Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C. Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R. McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and Patrick J. Coles. Variational quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, August 2021.

[Cerezo *et al.*, 2021b] M. Cerezo, Akira Sone, Tyler Volkoff, Lukasz Cincio, and Patrick J. Coles. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nature Communications*, 12(1):1791, March 2021.

[Choi *et al.*, 2023] Hyeongrak Choi, Frederic T Chong, Dirk Englund, and Yongshan Ding. Fault tolerant non-Clifford state preparation for arbitrary rotations. *arXiv preprint arXiv:2303.17380*, 2023.

[Coyle *et al.*, 2024] Brian Coyle, El Amine Cherrat, Nishant Jain, Natansh Mathur, Snehal Raj, Skander Kazdaghli, and Iordanis Kerenidis. Training-efficient density quantum machine learning, May 2024. arXiv:2405.20237 [quant-ph].

[Ding *et al.*, 2024] Zhiyan Ding, Taehee Ko, Jiahao Yao, Lin Lin, and Xiantao Li. Random coordinate descent: A simple alternative for optimizing parameterized quantum circuits. *Physical Review Research*, 6(3), July 2024.

[Duffield *et al.*, 2023] Samuel Duffield, Gabriel Matos, and Melf Johannsen. qujax: Simulating quantum circuits with JAX. *Journal of Open Source Software*, 8(89):5504, September 2023.

[Farhi *et al.*, 2014] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, 2014.

[Glorot and Bengio, 2010] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 249–256, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010. PMLR.

[Golay, 1977] M. Golay. Sieves for low autocorrelation binary sequences. *IEEE Transactions on Information Theory*, 23(1):43–51, 1977.

[Guerreschi and Smelyanskiy, 2017] Gian Giacomo Guerreschi and Mikhail Smelyanskiy. Practical optimization for hybrid quantum-classical algorithms, 2017.

[Gujju *et al.*, 2024] Yaswitha Gujju, Atsushi Matsuo, and Rudy Raymond. Quantum machine learning on near-term quantum devices: Current state of supervised and unsupervised techniques for real-world applications. *Physical Review Applied*, 21(6):067001, June 2024.

[Hao *et al.*, 2024] Tianyi Hao, Zichang He, Ruslan Shaydulin, Jeffrey Larson, and Marco Pistoia. End-to-end protocol for high-quality QAOA parameters with few shots, 2024.

[Harrow and Napp, 2021] Aram W. Harrow and John C. Napp. Low-depth gradient measurements can improve convergence in variational hybrid quantum-classical algorithms. *Physical Review Letters*, 126(14), April 2021.

[He *et al.*, 2024] Zichang He, Ruslan Shaydulin, Dylan Herman, Changhao Li, Rudy Raymond, Shree Hari Sureshbabu, and Marco Pistoia. Parameter setting heuristics make the quantum approximate optimization algorithm suitable for the early fault-tolerant era. *arXiv preprint arXiv:2408.09538*, 2024.

[James *et al.*, 2013] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An Introduction to Statistical Learning: with Applications in R*. Springer, 2013.

[Khairy *et al.*, 2020] Sami Khairy, Ruslan Shaydulin, Lukasz Cincio, Yuri Alexeev, and Prasanna Balaprakash. Learning to optimize variational quantum circuits to solve combinatorial problems. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34 (03), pages 2367–2375, 2020.

[Kim *et al.*, 2023] Youngseok Kim, Andrew Eddins, Sajant Anand, Ken Xuan Wei, Ewout Van Den Berg, Sami Rosenblatt, Hasan Nayfeh, Yantao Wu, Michael Zaletel, Kristan Temme, and Abhinav Kandala. Evidence for the utility of quantum computing before fault tolerance. *Nature*, 618(7965):500–505, June 2023.

[Kobayashi *et al.*, 2022] Masahiro Kobayashi, Kouhei Nakaji, and Naoki Yamamoto. Overfitting in quantum machine learning and entangling dropout. *Quantum Machine Intelligence*, 4(2):30, November 2022.

[Kondo *et al.*, 2024] Ruho Kondo, Yuki Sato, Rudy Raymond, and Naoki Yamamoto. Recursive quantum relaxation for combinatorial optimization problems, 2024.

[Layden *et al.*, 2023] David Layden, Guglielmo Mazzola, Ryan V. Mishmash, Mario Motta, Pawel Wocjan, Jin-Sung Kim, and Sarah Sheldon. Quantum-enhanced Markov chain Monte Carlo. *Nature*, 619(7969):282–287, July 2023.

[Mari *et al.*, 2021] Andrea Mari, Thomas R. Bromley, and Nathan Killoran. Estimating the gradient and higher-order derivatives on quantum hardware. *Physical Review A*, 103(1), January 2021.

[McClean *et al.*, 2016] Jarrod R McClean, Jonathan Romero, Ryan Babbush, and Alán Aspuru-Guzik. The theory of variational hybrid quantum-classical algorithms. *New Journal of Physics*, 18(2):023023, February 2016.

[McClean *et al.*, 2018] Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babbush, and Hartmut Neven. Barren plateaus in quantum neural network training landscapes. *Nature communications*, 9(1):4812, 2018.

[Mitarai *et al.*, 2018] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii. Quantum circuit learning. *Phys. Rev. A*, 98:032309, Sep 2018.

[Morris and Lotshaw, 2024] Titus D Morris and Phillip C Lotshaw. Performant near-term quantum combinatorial optimization. *arXiv preprint arXiv:2404.16135*, 2024.

[Moses *et al.*, 2023] Steven A Moses, Charles H Baldwin, Michael S Allman, R Ancona, L Ascarrunz, C Barnes, J Bartolotta, B Bjork, P Blanchard, M Bohn, et al. A race-track trapped-ion quantum processor. *Physical Review X*, 13(4):041052, 2023.

[Nakanishi *et al.*, 2020] Ken M. Nakanishi, Keisuke Fujii, and Synge Todo. Sequential minimal optimization for quantum-classical hybrid algorithms. *Phys. Rev. Res.*, 2:043158, Oct 2020.

[Nielsen and Chuang, 2010] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.

[Niu *et al.*, 2019] Murphy Yuezhen Niu, Sirui Lu, and Isaac L. Chuang. Optimizing QAOA: Success probability and runtime dependence on circuit depth, 2019.

[Ostaszewski *et al.*, 2021] Mateusz Ostaszewski, Edward Grant, and Marcello Benedetti. Structure optimization for parameterized quantum circuits. *Quantum*, 5:391, January 2021.

[Otterbach *et al.*, 2017] J. S. Otterbach, R. Manenti, N. Alidoust, A. Bestwick, M. Block, B. Bloom, et al. Unsupervised machine learning on a hybrid quantum computer, 2017.

[Packebusch and Mertens, 2016] Tom Packebusch and Stephan Mertens. Low autocorrelation binary sequences. *Journal of Physics A: Mathematical and Theoretical*, 49(16):165001, mar 2016.

[Pan *et al.*, 2022] Yu Pan, Yifan Tong, and Yi Yang. Automatic depth optimization for quantum approximate optimization algorithm. *Physical Review A*, 105(3):032433, March 2022. arXiv:2206.14412 [quant-ph].

[Scala *et al.*, 2023] Francesco Scala, Andrea Ceschini, Massimo Panella, and Dario Gerace. A general approach to dropout in quantum neural networks. *Advanced Quantum Technologies*, n/a(n/a):2300220, 2023.

[Shalev-Shwartz *et al.*, 2017] Shai Shalev-Shwartz, Ohad Shamir, and Shaked Shammah. Failures of gradient-based deep learning. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 3067–3075, 2017.

[Shaydulin *et al.*, 2023] Ruslan Shaydulin, Phillip C Lotshaw, Jeffrey Larson, James Ostrowski, and Travis S Humble. Parameter transfer for quantum approximate optimization of weighted MaxCut. *ACM Transactions on Quantum Computing*, 4(3):1–15, 2023.

[Shaydulin *et al.*, 2024] Ruslan Shaydulin, Changhao Li, Shouvanik Chakrabarti, Matthew DeCross, Dylan Herman, et al. Evidence of scaling advantage for the quantum approximate optimization algorithm on a classically intractable problem. *Science Advances*, 10(22), May 2024.

[Simon *et al.*, 2013] Noah Simon, Jerome Friedman, Trevor Hastie, and Robert Tibshirani. A sparse-group lasso. *Journal of Computational and Graphical Statistics*, 22(2):231–245, 2013.

[Sureshbabu *et al.*, 2024] Shree Hari Sureshbabu, Dylan Herman, Ruslan Shaydulin, Joao Basso, Shouvanik Chakrabarti, Yue Sun, and Marco Pistoia. Parameter setting in quantum approximate optimization of weighted problems. *Quantum*, 8:1231, 2024.

[Sweke *et al.*, 2020] Ryan Sweke, Frederik Wilde, Johannes Meyer, Maria Schuld, Paul K. Faehrmann, Barthélémy Meynard-Piganeau, and Jens Eisert. Stochastic gradient descent for hybrid quantum-classical optimization. *Quantum*, 4:314, August 2020.

[Tibshirani, 1996] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

[von Luxburg and Schoelkopf, 2008] Ulrike von Luxburg and Bernhard Schoelkopf. Statistical learning theory: Models, concepts, and results, 2008.

[Wada *et al.*, 2024] Kaito Wada, Rudy Raymond, Yuki Sato, and Hiroshi C Watanabe. Sequential optimal selections of single-qubit gates in parameterized quantum circuits. *Quantum Science and Technology*, 9(3):035030, may 2024.

[Wang *et al.*, 2018] Zhihui Wang, Stuart Hadfield, Zhang Jiang, and Eleanor G. Rieffel. Quantum approximate optimization algorithm for MaxCut: A fermionic view. *Physical Review A*, 97(2), February 2018.

[Wang *et al.*, 2021] Samson Wang, Enrico Fontana, M. Cerezo, Kunal Sharma, Akira Sone, Lukasz Cincio, and Patrick J. Coles. Noise-induced barren plateaus in variational quantum algorithms. *Nature Communications*, 12(1):6961, November 2021.

[Wauters *et al.*, 2020] Matteo M. Wauters, Emanuele Panizon, Glen B. Mbeng, and Giuseppe E. Santoro. Reinforcement-learning-assisted quantum optimization. *Physical Review Research*, 2(3), September 2020.

[Wierichs *et al.*, 2022] David Wierichs, Josh Izaac, Cody Wang, and Cedric Yen-Yu Lin. General parameter-shift rules for quantum gradients. *Quantum*, 6:677, March 2022.

[Wright, 2015] Stephen J. Wright. Coordinate descent algorithms. *Math. Program.*, 151(1):3–34, 2015.

[Yao *et al.*, 2020] Jiahao Yao, Marin Bukov, and Lin Lin. Policy gradient based quantum approximate optimization algorithm. In Jianfeng Lu and Rachel Ward, editors, *Proceedings of The First Mathematical and Scientific Machine Learning Conference*, volume 107 of *Proceedings of Machine Learning Research*, pages 605–634. PMLR, 20–24 Jul 2020.

[Zhou *et al.*, 2020] Leo Zhou, Sheng-Tao Wang, Soonwon Choi, Hannes Pichler, and Mikhail D Lukin. Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices. *Physical Review X*, 10(2):021067, 2020.