Submitted to 3rd Annual Conference on Mathematical and Scientific Machine Learning (MSML 2022)

Monte Carlo Tree Search based Hybrid Optimization of Variational Quantum Circuits

Jiahao Yao* JIAHAOYAO@BERKELEY.EDU
Department of Mathematics, University of California, Berkeley
Berkeley, CA 94720, USA

Haoya Li* LIHAOYA@STANFORD.EDU
Department of Mathematics, Stanford University
Stanford, CA, 94305, USA

Marin Bukov
Department of Physics, St. Kliment Ohridski University of Sofia
5 James Bourchier Blvd, 1164 Sofia, Bulgaria
Max Planck Institute for the Physics of Complex Systems, Nöthnitzer Str. 38, 01187 Dresden, Germany

Lin Lin
Department of Mathematics, University of California, Berkeley
Computational Research Division, Lawrence Berkeley National Laboratory
Challenge Institute for Quantum Computation, University of California, Berkeley
Berkeley, CA 94720, USA

Lexing Ying
Department of Mathematics, Stanford University
Stanford, CA 94305, USA

Abstract

Variational quantum algorithms stand at the forefront of simulations on near-term and future fault-tolerant quantum devices. While most variational quantum algorithms involve only continuous optimization variables, the representational power of the variational ansatz can sometimes be significantly enhanced by adding certain discrete optimization variables, as is exemplified by the generalized quantum approximate optimization algorithm (QAOA). However, the hybrid discrete-continuous optimization problem in the generalized QAOA poses a challenge to the optimization. We propose a new algorithm called MCTS-QAOA, which combines a Monte Carlo tree search method with an improved natural policy gradient solver to optimize the discrete and continuous variables in the quantum circuit, respectively. We find that MCTS-QAOA has excellent noise-resilience properties and outperforms prior algorithms in challenging instances of the generalized QAOA.

* J.Y. & H.L. contributed equally

© J. Yao, H. Li, M. Bukov, L. Lin & L. Ying.

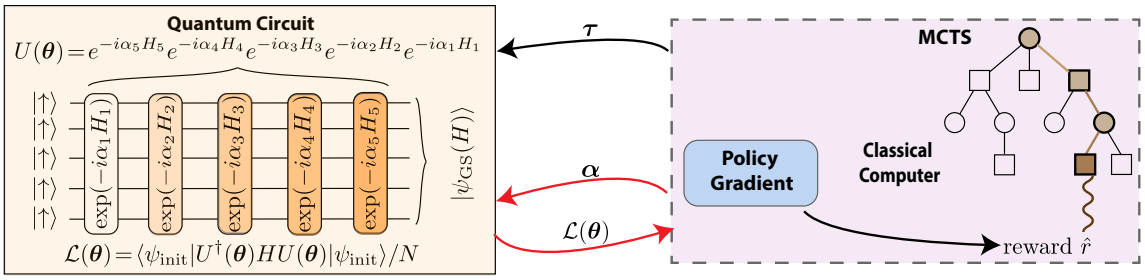

Figure 1: **The schematics of MCTS-QAOA**: MCTS provides promising paths for the discrete optimization search; the inner loop (highlighted in red) Policy Gradient (PG) solver evaluates the discrete sequence in a noise-robust way; the reward obtained is then propagated back through the search tree and used to improve the tree policy.

1. Introduction

Quantum computing provides a fundamentally different way for solving a variety of important problems in scientific computing, such as finding the ground state energy in computational chemistry, and the MaxCut problem in combinatorial optimization. Variational quantum circuits are perhaps the most important quantum algorithms on near term quantum devices (Preskill, 2018), mainly due to the tunability and the relatively short circuit depth (Cerezo et al., 2021b), as exemplified by the variational quantum eigensolver (VQE) (Peruzzo et al., 2014; McClean et al., 2016) and the quantum approximate optimization algorithm (QAOA) (Farhi et al., 2014). A common thread in these algorithms is to variationally optimize a parameterized quantum circuit using classical methods to obtain an approximate ground state. For instance, in combinatorial optimization, QAOA encodes the classical objective function into a quantum Hamiltonian, and constructs a quantum circuit with a set of two alternating quantum gates. The continuous adjustable parameters are the duration or phases of the gates.

For quantum many-body problems, the expressivity of the QAOA ansatz may be limited: the exponentially large (in the number of qubits) Hilbert space may not be efficiently navigated by the dynamics generated by the alternating gate sequence. This can lead to circuit depths that grow with the system size (Ho and Hsieh, 2019), or render the target ground state outside the scope of accessible states altogether, thus fundamentally precluding its preparation. To address these problems, various versions of a generalized QAOA ansatz have been presented in recent works (Zhu et al., 2020; Yao et al., 2020c; Chandarana et al., 2021), where additional control Hamiltonians are used to generate the variational circuits. In general, these Hamiltonians are tailored to the many-body system whose ground state we seek to prepare, and the extended Hamiltonian pool is often constructed using ideas from variational counter-diabatic (CD) driving (Sels and Polkovnikov, 2017). When the optimization of the parameterized circuit is performed successfully, the generalized ansatz produces a closer approximation to the ground state than the original alternating QAOA ansatz. The generalized QAOA may also significantly reduce the total protocol duration T and therefore the depth of the quantum circuit while giving a high fidelity with respect to the ground state (Yao et al., 2020c).

However, the ansatz of the generalized QAOA also results in a more challenging optimization problem. The original QAOA only involves optimization of continuous parameters. The generalized QAOA ansatz, in contrast, leads to a hybrid optimization problem that involves both the discrete variables (the choice of quantum gates) and the continuous variables (the duration of each gate). To solve this hybrid optimization problem, we propose a novel algorithm that combines the Monte Carlo Tree Search (MCTS) algorithm (Coulom, 2006; Browne et al., 2012; Abramson, 2014; Silver et al., 2016, 2017) – a powerful method in exploring the discrete sequence, with an improved noise-robust natural policy gradient solver for the continuous variables of a fixed gate sequence.

Contributions:

- We propose the MCTS-QAOA algorithm which combines the MCTS algorithm and a noise-robust policy gradient solver. We show that it is not only efficient in exploring the quantum gate sequences but also robust in the presence of different types of noise.
- The proposed MCTS-QAOA algorithm produces accurate results for problems that appear difficult or infeasible for previous algorithms based on the generalized QAOA ansatz, such as RL-QAOA (Yao et al., 2020b). In particular, MCTS-QAOA shows superior performance in the large protocol duration regime, where the hybrid optimization becomes challenging.
- In order for the MCTS-QAOA algorithm to produce reliable optimal results, it is crucial that the inner loop solver finds the optimal continuous variables with high accuracy. Compared to the original PG-QAOA solver introduced in (Yao et al., 2020a), we improve the inner loop solver with entropy regularization and the natural gradient method, and implement it in Jax (Bradbury et al., 2018), which offers more accurate, stable, and efficiently computed solutions during the continuous optimization.
- For the physics models considered in this paper, we observe that there can be many "good" gate sequences. This means that for a large portion of gate sequences, the energy ratio obtained is not far away from the optimal energy ratio obtainable with the generalized QAOA ansatz, given that the continuous variables are solved with high quality. This phenomenon has not been recorded in the literature to the best of the authors' knowledge.

Related works:

Quantum control and variational quantum eigensolver: Traditional optimal quantum control methods, often used in prior works, are GRAPE (Khaneja et al., 2005) and CRAB (Caneva et al., 2011). More recently, success has been seen by the combination of traditional methods with machine learning (Schäfer et al., 2020; Wang et al., 2020a; Sauvage and Mintert, 2019; Fösel et al., 2020; Nautrup et al., 2019; Albarrán-Arriagada et al., 2018; Sim et al., 2021; Wu et al., 2020a,b; Anand et al., 2020; Dalgaard et al., 2022), and especially reinforcement learning (Niu et al., 2019; Fösel et al., 2018; August and Hernández-Lobato, 2018; Porotti et al., 2019; Wauters et al., 2020; Yao et al., 2020a; Sung, 2020; Chen et al., 2013; Bukov, 2018; Bukov et al., 2018; Sørdal and Bergli, 2019; Bolens and Heyl, 2020; Dalgaard et al., 2020; Metz and Bukov, 2022; Baba et al., 2022)). Among them, Variational quantum eigensolver or VQE (Cerezo et al., 2021a; Tilly et al., 2021) provides a general framework applicable on noisy intermediate-scale quantum (NISQ) devices (Preskill, 2018) to variationally tune the circuit parameters and improve the approximation. In

the fault tolerant setting, there are also possibilities of error mitigation via the variational quantum optimization (Sung et al., 2020; Arute et al., 2020).

QAOA (Farhi et al., 2014) can be viewed as a specific variational quantum algorithm, and can be extended to the generalized QAOA ansatz (Zhu et al., 2020; Yao et al., 2020c; Chandarana et al., 2021). Prior works optimize the generalized QAOA greedily and progressively for each circuit layer or end-to-end as a large autoregressive network. The present work differs from these methods; we take advantage of the MCTS structure and formulate the problem as a two-level optimization.

MCTS and RL: Monte Carlo tree search (MCTS) has been one major workhorse behind the recent breakthrough of reinforcement learning algorithm, especially AlphaGo algorithms and variants (Silver et al., 2016, 2017, 2018; Schrittwieser et al., 2020; Ye et al., 2021). MCTS (Browne et al., 2012; Guo et al., 2014) makes use of a discrete hierarchical structure to figure out a better exploration in high dimensional search problems. While it is typically applied to discrete search, it has also been used in the continuous setting (Wang et al., 2020b), where the partition space of the whole space is viewed as branching of the tree. In the context of quantum computing, applications of MCTS have been recently emerged such as the Quantum Circuit Transformation (Zhou et al., 2020b), the quantum annealing schedules (Chen et al., 2020), and the quantum dynamics optimization (Dalgaard et al., 2020).

Further related works in hybrid optimization, counter-diabatic driving methods, and architecture search can be found in Appendix A.

2. Generalized QAOA ansatz

The generalized QAOA ansatz (Yao et al., 2020c) constructs a variational quantum circuit via the composition of a sequence of parameterized unitary operators:

$$U(\boldsymbol{\theta}) = \prod_{j=1}^{q} U(\tau_j, \alpha_j) = \prod_{j=1}^{q} \exp\left(-i\alpha_j H_{\tau_j}\right). \tag{2.1}$$

Here the circuit parameters $\boldsymbol{\theta} = (\boldsymbol{\alpha}, \boldsymbol{\tau})$ contain two components: i) the *discrete* variables $\boldsymbol{\tau} = (\tau_1, \tau_2, \ldots, \tau_q)$ define a sequence of Hamiltonians with length q, while ii) the *continuous* variables $\boldsymbol{\alpha} = \{\alpha_j\}_{j=1}^{q}$ represent the duration that each corresponding gate is applied for. It is further assumed that each Hamiltonian H_{τ_j} is selected from a fixed Hamiltonian pool $\mathcal{A} = \{H_1, H_2, \cdots, H_{|\mathcal{A}|}\}$, and consecutive gates are not repeated, i.e., $\tau_j \neq \tau_{j+1},\ 1 \leq j \leq q-1$. The total number of possible sequences is thus $|\mathcal{A}|(|\mathcal{A}|-1)^{q-1}$, which grows exponentially with q, rendering exhaustive search intractable.

After applying the circuit to an initial quantum state $|\psi_{\text{init}}\rangle$, one obtains the final quantum state $|\psi\rangle = U(\boldsymbol{\theta})\,|\psi_{\text{init}}\rangle$. To prepare a high quality approximation of the ground state $|\psi_{\text{GS}}\rangle$ of the target Hamiltonian H, the continuous and discrete variables are solved for by minimizing the following objective function:

$$\mathcal{L}(\boldsymbol{\theta}) = E(\boldsymbol{\theta})/N = \langle\psi_{\text{init}}|U^{\dagger}(\boldsymbol{\theta})HU(\boldsymbol{\theta})|\psi_{\text{init}}\rangle/N. \tag{2.2}$$

Note that the energy E in the objective function is divided by the number of particles N in the physical model, e.g. the number of qubits. This scaled objective function has a well-behaved limit

when increasing the number of qubits, as required for larger-scale computations. Here, the energy function $E(\boldsymbol{\theta})$ is always lower-bounded by the ground state energy $E_{\text{GS}} = \langle\psi_{\text{GS}}|H|\psi_{\text{GS}}\rangle$. It is also worth noticing that the quantum states $|\psi\rangle$ are unknown to the optimization algorithm (they cannot be measured), which increases the difficulty of the optimization algorithm.

3. Reinforcement learning setup

After defining the optimization problem posed by the generalized QAOA, let us briefly cast it within the RL framework.

3.1. Quantum constraints on the RL environment

Beyond classical physics, quantum mechanics imposes counterintuitive constraints on the state and reward spaces, which need to be embedded in a realistic RL environment.

First, the quantum state (or wavefunction) is not a physical observable by itself, and inference of the information of the full quantum state from experiments (called quantum state tomography) can require exponential resources. This fact is intimately related to the expected superior performance of quantum computers against their classical counterparts on certain tasks. To embed this quantum behavior into our environment simulator, we define the RL state as the sequence of actions applied (Bukov, 2018) rather than the quantum state. Starting from a fixed initial state, the quantum state is uniquely determined (though still unmeasurable) by the Hamiltonian sequence applied.

Second, (strong) quantum measurements lead to a collapse of the quantum wavefunction. This means that, once a measurement has been performed, the state itself is irreversibly lost. Therefore, a second constraint for our quantum RL environment is the sparsity of rewards. Indeed, only after the RL episode comes to an end, can we measure the energy and obtain the reward. In Sec. 4, we exploit this fact to introduce MCTS into the algorithm which does not evaluate the protocol $\boldsymbol{\tau}$ during the construction of it. As a result, the evaluation is delegated to the noise-robust PG-QAOA solver.

3.2. The reinforcement learning environment

In the language of reinforcement learning (RL), the choice of quantum gates corresponds to the action of the learner, and the quantum circuit is completed after q actions, which marks the end of the RL session/episode. The reward signal is provided by the inner loop solver which aims to compute the lowest possible energy that can be reached by the fixed chosen gate sequence. To be more specific, the action space $\mathcal{A} = \{H_j : 1 \leq j \leq |\mathcal{A}|\}$ is a set of Hamiltonians; the state space $\mathcal{S} = \{(\tau_1, \tau_2, \ldots, \tau_t) : \tau_j \in \mathcal{A}, 0 \leq j \leq t, 1 \leq t \leq q\}$ is the set of sequences of Hamiltonians with length no larger than q. In particular, a session always starts with the empty sequence s_0, and ends with a state given by a Hamiltonian sequence of length q. When $s_t = (\tau_1, \tau_2, \ldots, \tau_t)$ is not a terminal state, i.e., $t < q$, the next state s_{t+1} is obtained by appending the $(t+1)$-th action τ_{t+1} at the end of s_t, i.e., $s_{t+1} = (\tau_1, \tau_2, \ldots, \tau_t, \tau_{t+1})$.

The reward $r(s)$ only depends on the state s, and it is set as 0 whenever s is not a terminal state. As explained in the previous section, this implements the physical constraint reflecting the

Algorithm 1 MCTS-QAOA

Input: UCB bound coefficient c, number of outer loop iterations T_{iter}, number of random initialization T_{init}.

Initialize the Monte Carlo tree.
for $t = 1, .., T_{\text{iter}}$ **do**
  Pick a node according to the tree policy π_{tree}, cf. Eq. (4.1), using the UCB bound with parameter c.
  if the tree node is not the terminal state **then**
    Randomly roll out from the current tree node to obtain a terminal state $\boldsymbol{\tau}_t$.
  end if
  for $i = 1, .., T_{\text{init}}$ **do**
    Run natural policy gradient method (see Algorithm 2) to obtain the estimated reward $r_t^{[i]}$.
  end for
  Choose the best gate sequence durations according to the maximum reward $\hat{r}_t = \max_i r_t^{[i]}$ across different random intialization of policy gradient.
  Back-propagate the reward $\hat{r}_t$ from the node up to the root and update the statistics (Q, N) on each node.
end for

inability to perform a strong quantum measurement without destroying the quantum state. When s is a terminal state $\boldsymbol{\tau} = (\tau_1, \tau_2, \ldots, \tau_q)$, we define

$$r(s) = r(\boldsymbol{\tau}) = -\min_{\alpha} E(\{\alpha_j\}_{j=1}^q, \boldsymbol{\tau})/N, \tag{3.1}$$

where $\{\alpha_j\}_{j=1}^q$ are the duration obtained by the inner loop continuous optimizer, and the energy E is defined in (2.2).

4. Monte Carlo tree search with improved policy gradient solver

In this section, we introduce MCTS-QAOA, an algorithm that solves the hybrid optimization problem defined by the generalized QAOA ansatz, using a combination of MCTS and an improved policy gradient solver. In the combined algorithm, MCTS serves as the solver for the outer optimization problem: it is used to search for high quality gate sequences $\boldsymbol{\tau}$. At the same time, we design an improved policy gradient solver to produce the optimal gates duration $\boldsymbol{\alpha}$ for the discrete sequence provided by MCTS. Finally, the outcome of the evaluation is propagated back through the nodes of the MC tree to improve the tree policy before the next iteration.

4.1. Discrete optimization: Monte Carlo tree search

MCTS-QAOA strikes an efficient balance between exploration and exploitation of the RL states, by leveraging the statistics recorded in a search tree. Each node of this tree corresponds to a state s; the child nodes denote all possible states s' following the state s. For the problem considered in this paper, trajectories are loop-free, since each child state s' has one more action attached than

its parent state s. Thus, we refer to a given node by its corresponding state. In particular, the root node corresponds to the empty state s_0, which has $|\mathcal{A}|$ children, one for each action; any other non-terminal state s has $|\mathcal{A}|-1$ children, reflecting the constraint that no action can follow itself, and a terminal state has none. During the search process, each node keeps track of the statistics of two quantities: i) $N(s,a)$ counts the selection of action a at state s; ii) $Q(s,a)$ is the expected reward after taking action a at state s. Intuitively, the average $Q(s,a)/N(s,a)$ is an estimate of how promising a child node is. Finally, a node s is called fully expanded, if all its children are visited in the search, i.e., if $N(s,a) \geq 1$ for all $a \in \mathcal{A}$; otherwise, s is called an expandable node, and is the focus of exploration.

In each MCTS iteration, the tree and the node statistics are updated as follows:

1. *Forming a search path.* Starting from the root node, if the current node is fully expanded, then one of its children is chosen according to the following Upper Confidence Bound (UCB) (Auer et al., 2002):

$$\pi_{\text{tree}}(s) = \arg\max_{a \in \mathcal{A}} \left(\frac{Q(s,a)}{N(s,a)} + c\sqrt{\frac{2\log N(s)}{N(s,a)}} \right), \tag{4.1}$$

   until reaching a terminal state or an expandable node; here $\pi_{\text{tree}}(s)$ denotes the tree policy. Then an unvisited child of the current node is chosen at random, unless the current node is a terminal state. After that, a simulation is rolled out with a uniform policy until reaching a terminal state.

2. *Evaluation and backup.* The reward $\hat{r}$[1] of the terminal state is evaluated by the inner loop solver and the tree statistics are updated using $Q(s,a) \leftarrow Q(s,a)+\hat{r}, N(s,a) \leftarrow N(s,a)+1$ for each visited edge (s,a).

For the generalized QAOA ansatz, the real challenge lies in the evaluation step. On the one hand, the overall minimization of the energy depends on the potential of the trajectory selected by the MCTS, whose role is to find the optimal trajectory sequence. On the other hand, if the accuracy of the evaluation is low, then the searching process can be stuck at a severely suboptimal solution. Similarly, if the evaluation is not efficient enough, then the benefit obtained by using quantum computation strategy will also be lost. And last but not least, if the evaluation results are not robust to noise, then the algorithm can hardly be carried out on quantum devices. Hence, the inner loop solver used to implement the evaluation must be able to efficiently offer high accuracy results while being robust to different kinds of noise. The above considerations refer to the generic case; in practice, the optimization dynamics of the algorithm is set by the properties of the optimization landscape.

4.2. Continuous optimization: natural policy gradient solver

For each terminal state $\boldsymbol{\tau} = (\tau_1, \tau_2, \ldots, \tau_q)$ reached in the MCTS process, an inner loop solver is invoked to produce the optimal duration $\boldsymbol{\alpha} = \{\alpha_j\}_{j=1}^q$ and the reward $-E(\{\alpha_j\}_{j=1}^q, \boldsymbol{\tau})/N$ which are then back-propagated through the tree to update the tree statistics. In order to ensure that the

1. In order to distinguish the estimated reward from the true reward $r(s)$ in the presence of noise, we denote the estimated reward as $\hat{r}$.

duration obtained has a practical magnitude and to allow for a fair comparison between algorithms, we further assume that the total duration of all gates is fixed as T, which can be seen as a protocol for the circuit depth.

The continuous optimization problem for the inner-loop solver in the reward-evaluation step is thus

$$\min_{\{\alpha_j\}_{j=1}^q} \left\{ E(\{\alpha_j\}_{j=1}^q, \boldsymbol{\tau}) : \sum_{j=1}^q \alpha_j = T;\ 0 \le \alpha_j \le T \right\}. \tag{4.2}$$

In order to avoid using explicit derivatives of the energy E, we instead optimize the expectation of the energy E over a parameterized probability distribution of $\boldsymbol{\alpha}$; this is also crucial to to make the algorithm resilient to noise. More specifically, we set $\alpha_j = \frac{T\tilde{\alpha}_j}{\sum_k \tilde{\alpha}_k}$ to ensure the constraints on α_j, where $\tilde{\alpha}_j$ is a random variable drawn from the sigmoid Gaussian distribution $\mathcal{SN}(\mu_j, \sigma_j)$[2]. It can be parameterized as $\tilde{\alpha}_j = \mathfrak{g}(\delta_j)$, where $\delta_j \sim \mathcal{N}(\mu_j, \sigma_j)$ is a Gaussian random variable and $\mathfrak{g}(x) = \frac{1}{1+\exp(-x)}$ is the sigmoid function. Adding a Shannon entropy regularizer to the total expected reward we obtain the regularized objective function:

$$\mathcal{J}(\{\mu_j, \sigma_j\}_{j=1}^q) = \mathbb{E}_{\delta_j \sim \mathcal{N}(\mu_j, \sigma_j)} \left[R(\boldsymbol{\delta}) \right] + \beta_S^{-1} \sum_{j=1}^q \log \sigma_j, \tag{4.3}$$

which is maximized over the parameters $\{\mu_j, \sigma_j\}_{j=1}^q$. Here $R(\boldsymbol{\delta}) = -E\left(\left\{ \frac{T\mathfrak{g}(\delta_j)}{\sum_k \mathfrak{g}(\delta_k)} \right\}_{j=1}^q, \boldsymbol{\tau} \right)/N$, and β_S^{-1} denotes the temperature, which controls the trade-off between exploration and exploitation: higher temperature β_S^{-1} leads to a larger weight on the entropy term, and thus encourages exploration, while smaller β_S^{-1} reduces exploration. The entropy term $\sum_{j=1}^q \log \sigma_j$ can be derived from the definition of Shannon entropy, cf. Appendix D.

The inner loop solver is then constructed with a natural policy gradient (NPG) method applied to the regularized objective function $\mathcal{J}$ using the natural gradient direction $F^{-1}\nabla\mathcal{J}$, where F is the Fisher information matrix for the joint distribution of $\{\delta_j\}_{j=1}^q$ and $\nabla\mathcal{J}$ is the gradient of $\mathcal{J}$ with respect to the parameters. This procedure is different from the solver established in PG-QAOA (Yao et al., 2020a), where the standard gradient is used to update the parameters and no regularization is used. Using independent standard normal variables ξ_j, the natural gradient direction can be approximated by unbiased estimators:

$$F_j^{-1} \begin{bmatrix} \frac{\partial \mathcal{J}}{\partial \mu_j} \\ \frac{\partial \mathcal{J}}{\partial \log \sigma_j} \end{bmatrix} \approx \begin{bmatrix} \sigma_j R(\boldsymbol{\delta}) \xi_j \\ \frac{1}{2} R(\boldsymbol{\delta})(\xi_j^2 - 1) + \frac{1}{2}\beta_S^{-1} \end{bmatrix}, \tag{4.4}$$

where $\delta_j = \mu_j + \sigma_j \xi_j$ and F_j is the j-th 2-by-2 diagonal block of the Fisher information matrix, since F is a block diagonal matrix, cf. Appendix D. In practice, we update $\log\sigma$ instead of σ to ensure the positivity of σ, and we use the average of the unbiased estimators in (4.4) within a batch of size M to give the approximation of the natural gradient direction.

The first term in the objective function $\mathcal{J}$ can also be viewed as a smoothed reward function obtained with Gaussian perturbation. The parameter $\{\sigma_j\}_{j=1}^q$ determines the distance between

2. $\mathcal{SN}(\mu, \sigma)$ denotes the sigmoid Gaussian distribution with parameters μ and σ, i.e., the distribution of the Gaussian random variable $\mathcal{N}(\mu, \sigma)$ under the sigmoid transformation. It is also called the logit-normal distribution

Algorithm 2 Improved policy gradient solver

Input: Action sequence $\boldsymbol{\tau}$, number of restarts R, batch size M, learning rates η_t, total number of iterations K, the number of evaluation repeats m, the total gate duration T, the initial temperature β_S^{-1}, the rate of temperature decrease $0 < \gamma_T < 1$.

1: Randomly initialize the mean $\{\mu_j\}_{j=1}^q$ and variance $\{\sigma_j\}_{j=1}^q$.
2: **for** $t = 1, .., R \times K$ **do**
3: Sample a batch of variables $\{\tilde{\alpha}_j^l\}_{j=1}^q, l = 1, 2, \cdots, M$ of size M from sigmoid Gaussian distributions $\mathcal{SN}(\mu_j, \sigma_j)$.
4: Normalize the generalized QAOA parameter $\alpha_j = T\tilde{\alpha}_j / \sum_i \tilde{\alpha}_i$.
5: Compute the approximate NPG direction using Eq. (4.4).
6: Update the parameters with the gradient and learning rate η_t.
7: **if** t mod $K = 0$ and $t < (R-1)K$ **then** $\beta_S^{-1} \leftarrow \gamma_T \beta_S^{-1}$.
8: **if** $t = (R-1)K$ **then** $\beta_S^{-1} \leftarrow 0$.
9: **end for**
10: Apply the circuit m times with gate sequence τ and durations $\left\{\frac{T\mathfrak{g}(\mu_j)}{\sum_i \mathfrak{g}(\mu_i)}\right\}_{j=1}^q$, collect the rewards $\{r_k\}_{k=1}^m$, and estimate the reward $\hat{r}$ by $\hat{r} = \frac{1}{m}\sum_{k=1}^m r_k$.

Output: The mean and variance parameters $\{\mu_j\}_{j=1}^q$ and $\{\sigma_j\}_{j=1}^q$; the estimated reward $\hat{r}$.

$\mathcal{J}(\mu_j, \sigma_j)$ and $E(\mu_j)$ (Nesterov and Spokoiny, 2017). If σ is too large, then $\mathcal{J}$ is far from E, and yields suboptimal solutions of μ_j since too much details are lost after the Gaussian smoothing. To avoid this, we propose to use a tempering technique (see for example (Klink et al., 2020; Abdolmaleki et al., 2018; Haarnoja et al., 2018, Sec. 5)). More specifically, after a certain number of NPG iterations, we reduce the temperature β_S^{-1}, and in the final stage of entropy adjustment (cf. line 10-12 in Algorithm 2), we discard the entropy term. In this way, the policy is less susceptible to highly suboptimal local maxima in the beginning of the inner loop optimization thanks to the entropy regularization. At the end of the optimization, the variance σ_j decreases, since the temperature is reduced and the algorithm is able to achieve a higher precision as the smoothed problem becomes a better approximation to the original one. As a result, many policy gradient updates can be saved compared to the original policy gradient method in (Yao et al., 2020a), and the quality of solutions is improved.

When the optimization by the inner loop solver is completed, the parameters $\{\mu_j\}_{j=1}^q$ are used to evaluate the reward to be back-propagated through the MC tree. More specifically, the gate sequence τ with duration $\left\{\frac{T\mathfrak{g}(\mu_j)}{\sum_i \mathfrak{g}(\mu_i)}\right\}_{j=1}^q$ is applied and a reward is obtained. In order to deal with noisy rewards, the evaluation is repeated m times, and the average reward is sent to the discrete solver. The details of the inner loop algorithm is summarized in Algorithm 2.

4.3. Relation to previous algorithms used to optimize the generalized QAOA ansatz

We finish this section by a comparison of MCTS-QAOA with previous methods solving the QAOA problem. As shown in Table 1, the CD-QAOA method adopts Scipy solver for the continuous optimization, which cannot be applied to problems with noise, and the RL-QAOA method can produce suboptimal solutions in certain regimes, which we verify with numerical experiments in

Method	CD-QAOA	RL-QAOA	MCTS-QAOA
optimization (discrete)	AutoReg+PG	AutoReg+PG	MCTS
optimization (continuous)	SciPy		PG
performance without noise	✓	✗	✓
performance with noise	✗	✗	✓

Table 1: Comparison among the three algorithms for the generalized QAOA ansatz: CD-QAOA, RL-QAOA and MCTS-QAOA. In this table, AutoReg+PG stands for the policy gradient algorithm with the autoregressive neural network as a policy (Yao et al., 2020b); ✗means the algorithm can fail in certain challenging regimes (e.g., large total duration T).

the next section. Moreover, note that, due to the large neural network used in RL-QAOA, it is infeasible to apply the natural gradient methods as in Section 4.2.

5. Numerical experiments

To benchmark the performance of MCTS-QAOA, we consider three physics models: the 1-dimensional Ising model, the 2-dimensional Ising model on a square lattice, and the Lipkin-Meshkov-Glick (LMG) model. The description of the models and the additional Hamiltonians inspired from the counter-diabatic theory can be found in Appendix B. In addition, in order to test the noise-resilience of MCTS-QAOA, we consider three types of noise models: classical measurement Gaussian noise, quantum measurement noise, and gate rotation error, cf. Appendix C.

We compare the performance of MCTS-QAOA with that of RL-QAOA, and provide an analysis on why RL-QAOA might fail in certain regimes. Further analysis of the energy landscape of the discrete optimization reveals a surprising phenomenon: for generalized QAOA with optimal choices of the continuous degrees of freedom, there can be a large number of discrete protocols producing relatively accurate energies.

5.1. Comparison with RL-QAOA

For the methods solving the generalized QAOA problem summarized in Table 1, the CD-QAOA algorithm cannot be applied to problems with noise since the continuous solver is not noise-resilient, while the RL-QAOA algorithm has been shown to be effective with relatively short total duration JT (using unnormalized Hamiltonians (Yao et al., 2020b)). Therefore, we use RL-QAOA as a baseline when evaluating the performance of MCTS-QAOA, and we focus on the more challenging regime of large JT with normalized Hamiltonians[3].

3. The Hamiltonians used in this work are normalized by their operator norm $\|H\|$, i.e., we use $H/\|H\|$ instead of the original Hamiltonian H. The reason for introducing the normalized Hamiltonian is that the dependence of the cost of performing a Hamiltonian evolution $e^{-iH\alpha}$ on a quantum device – $\Omega(\|H\|\alpha)$ – scales with the norm (Berry et al., 2007; Low and Chuang, 2017). Interested readers can refer to Appendix B for more details.

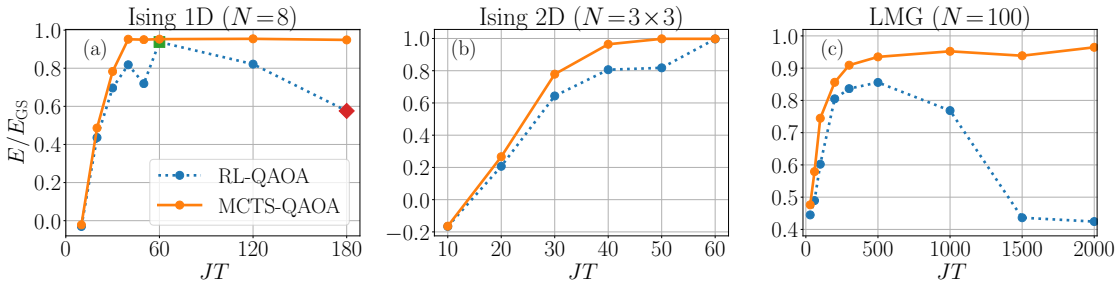

Figure 2: (**Quantum noise experiment**) comparison between MCTS-QAOA and RL-QAOA with quantum measurement noise (Appendix C). (a): 1D spin-1/2 Ising chain ($N = 8$) at $h_z/J=0.4523$ and $h_x/J=0.4045$; (b): 2D spin-1/2 Ising chain ($N=3\times3$) at $h_z/J=2$ and $h_x/J=3$; (c): LMG model ($N=100$) at $h/J = 0.9$. (see Sec. B for more details.) The blue dotted line and the orange solid line display the energy ratio $E/E_{\rm GS}$ obtained by RL-QAOA and MCTS-QAOA. The green square shape and the red diamond shape in the left panel approximately corresponds to $JT=10$ (an example in the small T regime) and $JT = 28$ (an example in the large T regime) with unnormalized Hamiltonians, respectively. The horizontal axis represents the total duration JT. MCTS-QAOA outperforms RL-QAOA in all tests.

We first compare the performance of MCTS-QAOA against that of RL-QAOA for the physical systems discussed in Appendix B in the presence of quantum noise. Detailed numerical results for the noiseless experiments and other noise models can be found in Appendix C. In order to compare the performance of different optimizers, noisy rewards are offered to the optimizers during the training process, and the exact rewards are only used in evaluating the protocols found by the optimizers. For MCTS-QAOA, the protocol evaluated is given by a greedy search, i.e., a searching process with the exploration coefficient $c = 0$ in Eq. (4.1).

Figure 2 shows the energy ratio evaluated for the protocols obtained by the optimizers across different lengths of total duration JT. For all three physics models, we find that the performance of MCTS-QAOA is at least as good as that of RL-QAOA for all protocol durations. In particular, for the 1D Ising model, MCTS-QAOA gives protocols that find close approximations to the true ground state when $JT \gtrsim 40$; RL-QAOA gives inferior solutions in these settings. For the 2D Ising model, while the performance of RL-QAOA is similar with that of MCTS-QAOA at $JT = 60$, the performance of RLQAOA at $JT = 30, 40$ and 50 is still inferior to that of MCTS-QAOA. For the LMG model, the quality of the gate sequence found by RL-QAOA further decreases when $JT > 500$, and MCTS-QAOA is significantly more robust.

The inferior performance of RL-QAOA is directly related to the joint parameterization used in RL-QAOA for the continuous and discrete policies. Since RL-QAOA optimizes the continuous and discrete variables simultaneously, for each discrete sequence, the level of accuracy of the continuous optimization can be relatively low. Consequently, the optimizer can get stuck at a suboptimal discrete sequence.

To illustrate this behavior, we analyze the training of RL-QAOA using the LMG model with $(JT, N, q) = (1500, 100, 8)$ and noiseless rewards. Figure 3 summarizes the performance of RL-

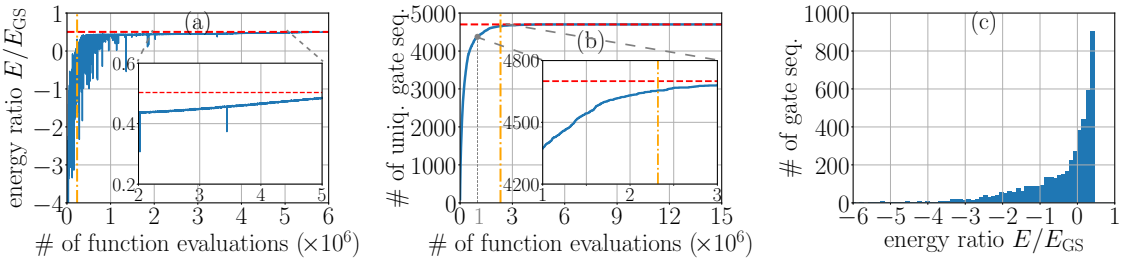

Figure 3: **Analysis of RL-QAOA using the LMG test**: (a): Energy ratio versus number of function evaluations; (b): Number of unique gate sequences encountered versus number of function evaluations; (c): Histogram of the rewards received by the algorithm in the first 5000 iterations. The horizontal red line in the left / middle panel represents the maximal energy ratio and the maximal number of unique gate sequences encountered during the optimization, respectively. The orange line marks the transition between two stages of the training process.

QAOA. Here by function evaluation we mean the computation of the objective function in (2.2). From Figure 3(a) and Figure 3(b), it is clear that the training process can be divided into two distinct stages, and the transition between the two stages is marked by the dashed-dotted vertical lines[4]. In stage I, which is to the left of the vertical lines, the number of unique gate sequences encountered by RL-QAOA quickly increases, while the energy ratio keeps oscillating below zero, which suggests that RL-QAOA focuses on exploration and the continuous optimization is done only very roughly within stage I. In stage II, which is to the right of the vertical lines, the number of unique gate sequences encountered by RL-QAOA stops to grow, while the energy ratio obtained grow above zero and eventually gets stuck at around 0.5, which means that the algorithm stops its exploration and focuses on the optimization of the continuous variables for a fixed gate sequence with stage II. The overall performance of RL-QAOA can highly depend on the discrete gate sequence that the RL-QAOA agent decides to exploit. In the next section, we demonstrate that both the exploration and the exploitation phases in RL-QAOA can be suboptimal in this example, but the main issue is related to the suboptimal discrete sequences found in the exploration phase.

5.2. Landscape of the discrete optimization and comparison with random search

In order to further understand the relative importance of continuous optimization versus discrete optimization for the generalized QAOA, we study the energy landscape of discrete optimization. For each discrete gate sequence, we perform numerical optimization to identify the *best* continuous parameters $\{\alpha_j\}$, and record the corresponding energy ratio.

Energy landscape of discrete optimization. – A profile of the discrete optimization landscape can be given by solving the corresponding continuous optimization individually on a random subset of all possible gate sequences; if the total number of possible gate sequences $|\mathcal{A}|(|\mathcal{A}|-1)^{q-1}$

4. These vertical lines are drawn at the point where the number of discrete protocol gate sequences drops to 10% of the total number within a single mini-batch

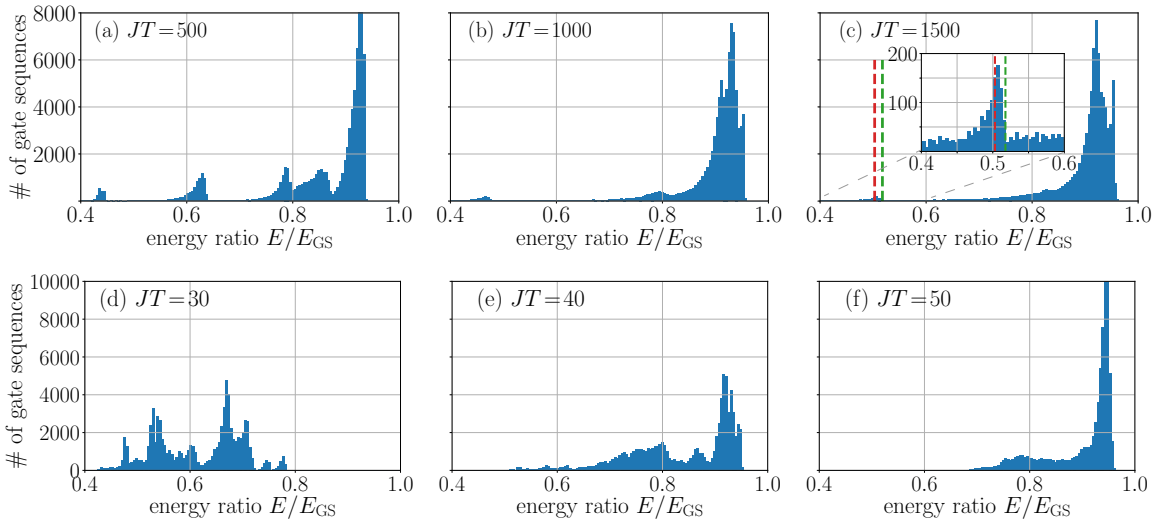

Figure 4: **Discrete landscape of the LMG model (a, b, c) and the 1D Ising model (d, e, f)**: Histograms of the energy ratio optimized by the improved natural gradient solver for $JT = 500, 1000, 1500$, respectively. $N_{\mathrm{hist}} = 81920$ samples are chosen from the discrete gate sequences of generalized QAOA with parameters $q = 8$, $|\mathcal{A}| = 5$ and $N = 100$ (LMG) or $N = 8$ (1D Ising). The dashed red line in the top right panel shows the energy ratio achieved by RL-QAOA in Figure 3; the green dashed line shows the energy ratio obtained by the NPG solver for the same gate sequences.

is relatively small, this subset can actually be chosen to include all sequences. In our numerical implementation, each discrete gate sequence is sent to the natural policy gradient solver described in Section 4.2, and the continuous variables are solved for different JT regime. Histograms for the energy ratios obtained can then be drawn.

Figure 4 shows the discrete landscape for the LMG model and the 1D Ising model, respectively, where the parameters of the ansatz are $(|\mathcal{A}|, q) = (5, 8)$, and the total number of gate sequences is thus 81920. From the histogram plot, most gate sequences are concentrated at the right-most peak in the large JT regime. Far from searching for "a needle in a haystack", this showcases that there are plenty of "good"[5] gate sequences assuming that each continuous optimization parameter is well solved. Note that the behavior is significantly different from the discrete-only optimization, where the landscape has been shown to feature transitions between glassy, correlated and uncorrelated phases (Day et al., 2019). To the best of our knowledge, the existence of many good discrete gate sequences in the QAOA-type variational quantum algorithms has not been reported in the literature.

For the LMG model with total gate duration $JT = 1500$ (cf. Figure 4), while most energy ratios fall into the cluster above 0.9, there is a smaller cluster located at 0.5. The energy ratio obtained by RL-QAOA (cf. Figure 3) falls into this cluster, which is depicted by the red dashed line, while the

5. "Good" gate sequences here means the optimized energy ratio is close to the optimal energy ratio obtainable within the generalized QAOA ansatz.

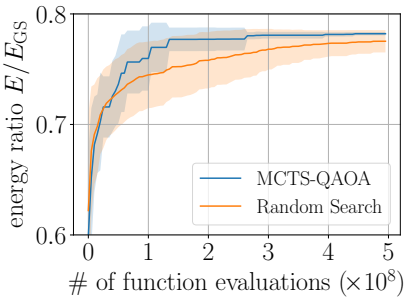

Figure 5: **Comparison between MCTS-QAOA and Random Search**: The blue and orange curves correspond to MCTS-QAOA and random search, respectively. The physics system is the 1D Ising model with duration $JT = 30$, which corresponds to Figure 2 (a). The generalized QAOA parameters are $q = 8$ and $|\mathcal{A}| = 5$. The horizontal axis is the number of function evaluations (with the function evaluation in the continuous optimization taken into account), and the vertical axis is the energy ratio. The shaded area for both algorithms represents the standard deviation across ten different random initializations.

green dashed line shows the energy ratio obtained by the natural policy gradient solver with the same gate sequence. The green line corresponds to a higher energy ratio than the red line, which means that the optimization of the continuous variables in the second stage of RL-QAOA is not as good as the NPG solver, and the difference between the two lines indicates the suboptimality caused by the exploitation. However, the suboptimality of the RL-QAOA solution is mainly due to the exploration stage, since the discrete sequence that RL-QAOA chooses to exploit represents a suboptimal local optimum that belongs to a cluster much smaller than the rightmost one in the histogram. The top right panel of Figure 4 also verifies the claim that RL-QAOA only does a rough optimization on the continuous variables before it stops exploration, since the energy ratios displayed there are mostly above 0.4, while the energy ratio obtained in optimization stage I is mainly negative. While the landscape of the hybrid optimization is challenging for RL-QAOA, the proposed method MCTS-QAOA is able to deal with it by using a noise-resilient solver for the continuous variables (NPG), and by exploring the discrete variables constantly using MCTS.

For the 1D Ising model with total gate duration $JT = 30$ shown in the bottom left panel, where the rightmost cluster is not the largest. This means that in this setting, it is more difficult to find a gate sequence in the rightmost cluster when the random search is used. We examine the performance of random search and MCTS-QAOA using this example in the next part.

Comparison with random search. – A recent work (Mania et al., 2018) points out that advanced RL methods *need not* outperform simpler methods such as random search. In fact, if there is no specific structure in a problem, a random search algorithm might be as efficient as any sophisticated algorithm. In addition, from the landscape illustrated in the previous histograms, one can see that, *for the models we investigated*, there are lots of gate sequences with relatively high energy ratios *provided that* the continuous protocols are optimized. Therefore, it is natural to com-

pare MCTS-QAOA against the random search algorithm[6]. For a fair comparison, we assume that the continuous optimization in both cases is solved by the natural policy gradient algorithm, and the difference only lies in the discrete optimization. In Figure 5, the best energy ratio in the training history is shown for the two methods, and one sees that MCTS-QAOA consistently outperforms the random search across different random seeds. MCTS-QAOA not only finds better gate sequences much faster, but also gives a smaller variance across different realizations. It is clear that instead of doing the search uniformly and treating each protocol as equally important, the tree statistics in MCTS-QAOA better guides into a more promising search direction.

6. Conclusion and discussions

In this paper, we study a continuous-discrete variational quantum algorithm for the generalized QAOA ansatz. To solve this hybrid optimization problem, we design a novel algorithm that combines the Monte Carlo tree search (MCTS) algorithm, a powerful method in exploring the discrete sequence, with an improved noise-robust policy gradient solver for the continuous duration variables of a fixed gate sequence. The proposed algorithms effectively generate robust quantum control where the prior methods fail.

In this context, we expect that random search algorithms cannot efficiently determine the best gate sequence if noisy rewards are used, while MCTS-QAOA is able to mitigate the noise and provide robust choice of gate sequence with the help of the tree structure it maintains. Moreover, it is possible for MCTS-QAOA to further reduce the number of evaluations by assigning different number of iterations for different gate sequences, e.g., it can assign more iterations for the more promising gate sequences. Also, MCTS-QAOA allows for the application of transfer learning using the tree statistics, which is not possible for the random search.

There are a number of possible ways to extend the problem presented in this paper:

Learning based guided search. – MCTS can be possibly guided by a learned functional approximator, such as neural networks or tensor networks. We have also tried the implementation of AlphaZero in the same experimental settings. However, the neural network based method does not work better than the simple MCTS. We find that the value function mapping from the discrete gate sequences to the score was quite hard to learn. One reason might be that the continuous policy gradient will try the best to optimize the energy ratio to the highest, thus making this mapping from discrete sequences to score, highly non-linear. Also, in terms of sampling efficiency, the neural network based approach needs lots of samples to fit the function, which is a heavy overhead compared to the simple MCTS approach. Nevertheless, the question remains open as to how to upgrade MCTS to a guided search.

Amortized computation. – The computation within the policy gradient solver for different gate sequences can possibly be amortized. Currently, the continuous and discrete optimizations are separated. If some functional can be learned by replaying the data during the policy gradient iteration, the number of function evaluations can be further reduced. However, one difficulty in the quantum

6. The random search algorithm also uses a two-level optimization, where the continuous optimization is solved by the policy gradient algorithm and the discrete optimization uses the random search. Since we assume no prior knowledge, the random search would be uniformly random on the discrete search space.

setting is that we do not have access to the quantum state, and thus we cannot learn a mapping taking the quantum state as input, unless we apply non-trivial quantum tomography. Therefore, how to reuse the past information and make the MCTS-QAOA algorithm quickly adaptive in physical setting remains to be investigated. Advanced algorithms like meta learning can be explored in the future work.

Budget-aware variational quantum algorithms. – A point of high interest is the design of budget-aware variational quantum algorithms. The importance of sample efficiency in the quantum setting can never be overemphasized. Each run of a quantum circuit can be expensive and quantum decoherence noise is usually not stationary over time. The budget-awareness property can be naturally incorporated in the MTCS framework. Making use of the tree structure, the adaptive algorithm would distribute more function evaluation budget to the most-visited or more promising nodes. The current algorithm likely operates in a budget-sufficient regime and uses the same amount of budget for each discrete gate sequences. We hope the adaptive algorithm can hit the sweet spot in the middle, i.e., use the right amount of computational budget and still compute the best possible gate sequence design. We hope that the present work will accelerate the research of budget-aware variational quantum algorithms in a realistic setting.

Acknowledgement

We thank Michael Luo for donating eight NVIDIA V100 Tensor Core GPUs to support computation. This work was partially supported by the Department of Energy under Grant No. DE-AC02-05CH11231 and No. DE-SC0017867 (L.L., J.Y.), and by the National Science Foundation under the NSF QLCI program through grant number OMA-2016245 (L.L.). M.B. was supported by the Marie Skłodowska-Curie grant agreement No 890711.

We used W&B (Biewald, 2020) to organize and analyze the experiments. The reinforcement learning networks are implemented in NumPy (Harris et al., 2020), and Jax (Bradbury et al., 2018); the quantum systems are simulated in Quspin (Weinberg and Bukov, 2017, 2019). We thank Berkeley Research Computing (BRC) and Google Cloud Computing Services (GCP) for providing the computational resources.

References

Abbas Abdolmaleki, Jost Tobias Springenberg, Yuval Tassa, Remi Munos, Nicolas Heess, and Martin Riedmiller. Maximum a posteriori policy optimisation. *arXiv preprint arXiv:1806.06920v1*, Jun 2018. URL http://arxiv.org/abs/1806.06920v1.

Bruce Abramson. *The expected-outcome model of two-player games*. Morgan Kaufmann, 2014.

Francisco Albarrán-Arriagada, Juan C Retamal, Enrique Solano, and Lucas Lamata. Measurement-based adaptation protocol with quantum reinforcement learning. *Phys. Rev. A*, 98:042315, 2018. doi: 10.1103/PhysRevA.98.042315. URL https://link.aps.org/doi/10.1103/PhysRevA.98.042315.

Abhinav Anand, Matthias Degroote, and Alan Aspuru-Guzik. Natural evolutionary strategies for variational quantum computation. *arXiv preprint arXiv:2012.00101*, 2020.

Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Sergio Boixo, Michael Broughton, Bob B Buckley, David A Buell, et al. Hartree-fock on a superconducting qubit quantum computer. *Science*, 369(6507):1084–1089, 2020.

Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

Moritz August and José Miguel Hernández-Lobato. Taking gradients through experiments: Lstms and memory proximal policy optimization for black-box quantum control. In *International Conference on High Performance Computing*, pages 591–613. Springer, 2018. URL https://arxiv.org/abs/1802.04063.

Shotaro Z. Baba, Nobuyuki Yoshioka, Yuto Ashida, and Takahiro Sagawa. Deep reinforcement learning for preparation of thermal and prethermal quantum states. *arXiv preprint arXiv:2207.12656v2*, Jul 2022. URL http://arxiv.org/abs/2207.12656v2.

Dominic W Berry, Graeme Ahokas, Richard Cleve, and Barry C Sanders. Efficient quantum algorithms for simulating sparse Hamiltonians. *Commun. Math. Phys.*, 270(2):359–371, 2007.

Dimitri Bertsekas. *Reinforcement learning and optimal control*. Athena Scientific, 2019.

Lukas Biewald. Experiment tracking with weights and biases, 2020. URL https://www.wandb.com/. Software available from wandb.com.

Adrien Bolens and Markus Heyl. Reinforcement learning for digital quantum simulation. *arXiv preprint arXiv:2006.16269*, 2020. URL https://arxiv.org/abs/2006.16269.

R Botet and R Jullien. Large-size critical behavior of infinitely coordinated systems. *Physical Review B*, 28(7):3955, 1983.

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. URL http://github.com/google/jax.

Cameron B Browne, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter I Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in games*, 4(1):1–43, 2012.

Marin Bukov. Reinforcement learning for autonomous preparation of floquet-engineered states: Inverting the quantum kapitza oscillator. *Physical Review B*, 98(22):224305, 2018. doi: 10.1103/PhysRevB.98.224305. URL https://link.aps.org/doi/10.1103/PhysRevB.98.224305.

Marin Bukov, Alexandre GR Day, Dries Sels, Phillip Weinberg, Anatoli Polkovnikov, and Pankaj Mehta. Reinforcement learning in different phases of quantum control. *Physical Review X*, 8(3): 031086, 2018. doi: 10.1103/PhysRevX.8.031086. URL https://link.aps.org/doi/10.1103/PhysRevX.8.031086.

Han Cai, Ligeng Zhu, and Song Han. Proxylessnas: Direct neural architecture search on target task and hardware. *arXiv preprint arXiv:1812.00332*, 2018.

Tommaso Caneva, Tommaso Calarco, and Simone Montangero. Chopped random-basis quantum optimization. *Phys. Rev. A*, 84:022326, 2011. doi: 10.1103/PhysRevA.84.022326. URL https://link.aps.org/doi/10.1103/PhysRevA.84.022326.

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, 2021a. doi: 10.1038/s42254-021-00348-9. URL https://doi.org/10.1038/s42254-021-00348-9.

Marco Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, et al. Variational quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, 2021b.

P. Chandarana, N. N. Hegade, Koushik Paul, F. Albarrán-Arriagada, Enrique Solano, A. del Campo, and Xi Chen. Digitized-counterdiabatic quantum approximate optimization algorithm. *arXiv preprint arXiv:2107.02789v2*, Jul 2021. URL http://arxiv.org/abs/2107.02789v2.

Chunlin Chen, Daoyi Dong, Han-Xiong Li, Jian Chu, and Tzyh-Jong Tarn. Fidelity-based probabilistic q-learning for control of quantum systems. *IEEE transactions on neural networks and learning systems*, 25(5):920–933, 2013. doi: 10.1109/TNNLS.2013.2283574.

Yu-Qin Chen, Yu Chen, Chee-Kong Lee, Shengyu Zhang, and Chang-Yu Hsieh. Optimizing quantum annealing schedules: From monte carlo tree search to quantumzero. *arXiv preprint arXiv:2004.02836v2*, Apr 2020. URL http://arxiv.org/abs/2004.02836v2.

Rémi Coulom. Efficient selectivity and backup operators in monte-carlo tree search. In *International conference on computers and games*, pages 72–83. Springer, 2006.

Mogens Dalgaard, Felix Motzoi, Jens Jakob Sørensen, and Jacob Sherson. Global optimization of quantum dynamics with AlphaZero deep exploration. *npj Quantum Information*, 6 (1), jan 2020. doi: 10.1038/s41534-019-0241-0. URL https://doi.org/10.1038%2Fs41534-019-0241-0.

Mogens Dalgaard, Felix Motzoi, and Jacob Sherson. Predicting quantum dynamical cost landscapes with deep learning. *Physical Review A*, 105(1):012402, 2022.

Alexandre GR Day, Marin Bukov, Phillip Weinberg, Pankaj Mehta, and Dries Sels. Glassy phase of optimal quantum control. *Physical review letters*, 122(2):020601, 2019. doi: 10.1103/PhysRevLett.122.020601.

Olivier Delalleau, Maxim Peter, Eloi Alonso, and Adrien Logut. Discrete and continuous action representation for practical rl in video games. *arXiv preprint arXiv:1912.11077*, 2019.

Thomas Elsken, Jan Hendrik Metzen, and Frank Hutter. Neural architecture search: A survey. *arXiv preprint arXiv:1808.05377v3*, Aug 2018. URL http://arxiv.org/abs/1808.05377v3. Journal of Machine Learning Research 20 (2019) 1-21.

Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A Quantum Approximate Optimization Algorithm. *arXiv preprint arXiv:1411.4028*, 2014. URL https://arxiv.org/pdf/1411.4028.pdf.

Thomas Fösel, Petru Tighineanu, Talitha Weiss, and Florian Marquardt. Reinforcement learning with neural networks for quantum feedback. *Phys. Rev. X*, 8:031084, 2018. doi: 10.1103/PhysRevX.8.031084. URL https://link.aps.org/doi/10.1103/PhysRevX.8.031084.

Thomas Fösel, Stefan Krastanov, Florian Marquardt, and Liang Jiang. Efficient cavity control with snap gates. *arXiv preprint arXiv:2004.14256*, 2020. URL https://arxiv.org/abs/2004.14256.

Xiaoxiao Guo, Satinder Singh, Honglak Lee, Richard L Lewis, and Xiaoshi Wang. Deep learning for real-time atari game play using offline monte-carlo tree search planning. In *Advances in neural information processing systems*, pages 3338–3346, 2014.

Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2018. URL https://arxiv.org/abs/1812.05905.

Charles R. Harris, K. Jarrod Millman, St'efan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fern'andez del R'ıo, Mark Wiebe, Pearu Peterson, Pierre G'erard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, 2020. doi: 10.1038/s41586-020-2649-2. URL https://doi.org/10.1038/s41586-020-2649-2.

Andreas Hartmann and Wolfgang Lechner. Rapid counter-diabatic sweeps in lattice gauge adiabatic quantum computing. *New Journal of Physics*, 21(4):043025, 2019. doi: 10.1088/1367-2630/ab14a0. URL https://doi.org/10.1088%2F1367-2630%2Fab14a0.

N. N. Hegade, P. Chandarana, K. Paul, X. Chen, F. Albarrán-Arriagada, and E. Solano. Portfolio optimization with digitized-counterdiabatic quantum algorithms. *arXiv preprint arXiv:2112.08347v1*, Dec 2021a. URL http://arxiv.org/abs/2112.08347v1.

Narendra N. Hegade, Koushik Paul, Yongcheng Ding, Mikel Sanz, F. Albarrán-Arriagada, Enrique Solano, and Xi Chen. Shortcuts to adiabaticity in digitized adiabatic quantum computing. *Physical Review Applied*, 15(2), feb 2021b. doi: 10.1103/physrevapplied.15.024038. URL https://doi.org/10.1103%2Fphysrevapplied.15.024038.

Narendra N. Hegade, Xi Chen, and Enrique Solano. Digitized-counterdiabatic quantum optimization. *arXiv preprint arXiv:2201.00790v1*, Jan 2022. URL http://arxiv.org/abs/2201.00790v1.

Wen Wei Ho and Timothy H Hsieh. Efficient variational simulation of non-trivial quantum states. *SciPost Phys*, 6:29, 2019.

Navin Khaneja, Timo Reiss, Cindie Kehlet, Thomas Schulte-Herbrüggen, and Steffen J Glaser. Optimal control of coupled spin dynamics: design of nmr pulse sequences by gradient ascent algorithms. *Journal of magnetic resonance*, 172(2):296–305, 2005.

Hyungwon Kim and David A Huse. Ballistic spreading of entanglement in a diffusive nonintegrable system. *Physical review letters*, 111(12):127205, 2013.

Pascal Klink, Carlo D'Eramo, Jan R Peters, and Joni Pajarinen. Self-paced deep reinforcement learning. *Advances in Neural Information Processing Systems*, 33:9216–9227, 2020.

En-Jui Kuo, Yao-Lung L. Fang, and Samuel Yen-Chi Chen. Quantum architecture search via deep reinforcement learning. *arXiv preprint arXiv:2104.07715v1*, Apr 2021. URL http://arxiv.org/abs/2104.07715v1.

Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016. URL http://arxiv.org/abs/1509.02971.

Harry J Lipkin, N Meshkov, and AJ Glick. Validity of many-body approximation methods for a solvable model:(i). exact solutions and perturbation theory. *Nuclear Physics*, 62(2):188–198, 1965.

Hanxiao Liu, Karen Simonyan, and Yiming Yang. DARTS: differentiable architecture search. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL https://openreview.net/forum?id=S1eYHoC5FX.

Guang Hao Low and Isaac L. Chuang. Optimal hamiltonian simulation by quantum signal processing. *Phys. Rev. Lett.*, 118:010501, 2017.

Horia Mania, Aurelia Guy, and Benjamin Recht. Simple random search provides a competitive approach to reinforcement learning. *arXiv preprint arXiv:1803.07055v1*, Mar 2018. URL http://arxiv.org/abs/1803.07055v1.

Gabriel Matos, Sonika Johri, and Zlatko Papić. Quantifying the efficiency of state preparation via quantum variational eigensolvers. *PRX Quantum*, 2(1), 2021. doi: 10.1103/prxquantum.2.010309. URL https://doi.org/10.1103%2Fprxquantum.2.010309.

Jarrod R McClean, Jonathan Romero, Ryan Babbush, and Alán Aspuru-Guzik. The theory of variational hybrid quantum-classical algorithms. *New J. Phys.*, 18(2):023023, 2016.

Friederike Metz and Marin Bukov. Self-correcting quantum many-body control using reinforcement learning with tensor networks. *arXiv preprint arXiv:2201.11790*, 2022.

Hendrik Poulsen Nautrup, Nicolas Delfosse, Vedran Dunjko, Hans J Briegel, and Nicolai Friis. Optimizing quantum error correction codes with reinforcement learning. *Quantum*, 3:215, 2019. doi: 10.22331/q-2019-12-16-215.

Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566, 2017.

Michael Neunert, Abbas Abdolmaleki, Markus Wulfmeier, Thomas Lampe, Jost Tobias Springenberg, Roland Hafner, Francesco Romano, Jonas Buchli, Nicolas Heess, and Martin Riedmiller. Continuous-discrete reinforcement learning for hybrid control in robotics. *arXiv preprint arXiv:2001.00449*, 2020.

Murphy Yuezhen Niu, Sergio Boixo, Vadim N Smelyanskiy, and Hartmut Neven. Universal quantum control through deep reinforcement learning. *npj Quantum Information*, 5 (1):1–8, 2019. doi: 10.1038/s41534-019-0141-3. URL https://doi.org/10.1038/s41534-019-0141-3.

G. Passarelli, V. Cataudella, R. Fazio, and P. Lucignano. Counterdiabatic driving in the quantum annealing of the p-spin model: A variational approach. *Phys. Rev. Research*, 2:013283, Mar 2020. doi: 10.1103/PhysRevResearch.2.013283. URL https://link.aps.org/doi/10.1103/PhysRevResearch.2.013283.

Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J Love, Alán Aspuru-Guzik, and Jeremy L O'brien. A variational eigenvalue solver on a photonic quantum processor. *Nature communications*, 5:4213, 2014. doi: 10.1038/ncomms5213. URL https://doi.org/10.1038/ncomms5213.

Riccardo Porotti, Dario Tamascelli, Marcello Restelli, and Enrico Prati. Coherent transport of quantum states by deep reinforcement learning. *Communications Physics*, 2(1):1–9, 2019. doi: 10.1038/s42005-019-0169-x. URL https://doi.org/10.1038/s42005-019-0169-x.

John Preskill. Quantum computing in the nisq era and beyond. *Quantum*, 2:79, 2018.

Esteban Real, Chen Liang, David So, and Quoc Le. Automl-zero: Evolving machine learning algorithms from scratch. In *International Conference on Machine Learning*, pages 8007–8019. PMLR, 2020.

Frederic Sauvage and Florian Mintert. Optimal quantum control with poor statistics. *arXiv preprint arXiv:1909.01229*, 2019. URL https://arxiv.org/abs/1909.01229.

Frank Schäfer, Michal Kloc, Christoph Bruder, and Niels Lörch. A differentiable programming method for quantum control. *Machine Learning: Science and Technology*, 1, 2020. doi: 10.1088/2632-2153/ab9802.

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, Timothy Lillicrap, and David Silver. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588 (7839):604–609, dec 2020. doi: 10.1038/s41586-020-03051-4. URL https://doi.org/10.1038%2Fs41586-020-03051-4.

Dries Sels and Anatoli Polkovnikov. Minimizing irreversible losses in quantum systems by local counterdiabatic driving. *Proceedings of the National Academy of Sciences*, 114(20):E3909–E3916, 2017.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018.

Sukin Sim, Jonathan Romero, Jérôme F Gonthier, and Alexander A Kunitsa. Adaptive pruning-based optimization of parameterized quantum circuits. *Quantum Science and Technology*, 6(2): 025019, mar 2021. doi: 10.1088/2058-9565/abe107. URL https://doi.org/10.1088%2F2058-9565%2Fabe107.

Vegard B Sørdal and Joakim Bergli. Deep reinforcement learning for robust quantum optimization. *arXiv preprint arXiv:1904.04712*, 2019. URL https://arxiv.org/abs/1904.04712.

Kevin Sung. *Towards the First Practical Applications of Quantum Computers*. PhD thesis, University of Michigan, 2020.

Kevin J Sung, Jiahao Yao, Matthew P Harrigan, Nicholas C Rubin, Zhang Jiang, Lin Lin, Ryan Babbush, and Jarrod R McClean. Using models to improve optimizers for variational quantum algorithms. *Quantum Science and Technology*, 5(4):044008, 2020.

Jules Tilly, Hongxiang Chen, Shuxiang Cao, Dario Picozzi, Kanav Setia, Ying Li, Edward Grant, Leonard Wossnig, Ivan Rungger, George H. Booth, and Jonathan Tennyson. The variational quantum eigensolver: a review of methods and best practices. *arXiv preprint arXiv:2111.05176v1*, Nov 2021. URL http://arxiv.org/abs/2111.05176v1.

Brandon Trabucco, Aviral Kumar, Xinyang Geng, and Sergey Levine. Conservative objective models for effective offline model-based optimization. In *International Conference on Machine Learning*, pages 10358–10368. PMLR, 2021.

Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, 575(7782):350–354, 2019.

Guoming Wang, Dax Enshan Koh, Peter D Johnson, and Yudong Cao. Bayesian inference with engineered likelihood functions for robust amplitude estimation. *arXiv preprint arXiv:2006.09350*, 2020a.

Hanrui Wang, Yongshan Ding, Jiaqi Gu, Zirui Li, Yujun Lin, David Z. Pan, Frederic T. Chong, and Song Han. Quantumnas: Noise-adaptive search for robust quantum circuits. *arXiv preprint arXiv:2107.10845v5*, Jul 2021a. URL http://arxiv.org/abs/2107.10845v5.

Hanrui Wang, Jiaqi Gu, Yongshan Ding, Zirui Li, Frederic T. Chong, David Z. Pan, and Song Han. Roqnn: Noise-aware training for robust quantum neural networks. *arXiv preprint arXiv:2110.11331v1*, Oct 2021b. URL http://arxiv.org/abs/2110.11331v1.

Linnan Wang, Rodrigo Fonseca, and Yuandong Tian. Learning search space partition for black-box optimization using monte carlo tree search. *arXiv preprint arXiv:2007.00708*, 2020b.

Matteo M Wauters, Emanuele Panizon, Glen B Mbeng, and Giuseppe E Santoro. Reinforcement learning assisted quantum optimization. *Phys. Rev. Research*, 2:033446, 2020. doi: 10.1103/PhysRevResearch.2.033446. URL https://link.aps.org/doi/10.1103/PhysRevResearch.2.033446.

Phillip Weinberg and Marin Bukov. Quspin: a python package for dynamics and exact diagonalisation of quantum many body systems part i: spin chains. *SciPost Phys*, 2(1), 2017.

Phillip Weinberg and Marin Bukov. Quspin: a python package for dynamics and exact diagonalisation of quantum many body systems. part ii: bosons, fermions and higher spins. *SciPost Phys.*, 7 (arXiv: 1804.06782):020, 2019.

Re-Bing Wu, Xi Cao, Pinchen Xie, and Yu-xi Liu. End-to-end quantum machine learning with quantum control systems. *arXiv preprint arXiv:2003.13658*, 2020a. URL https://arxiv.org/abs/2003.13658.

Yadong Wu, Zengming Meng, Kai Wen, Chengdong Mi, Jing Zhang, and Hui Zhai. Active learning approach to optimization of experimental control. *arXiv preprint arXiv:2003.11804*, 2020b.

Jonathan Wurtz and Peter J Love. Counterdiabaticity and the quantum approximate optimization algorithm. *arXiv preprint arXiv:2106.15645*, 2021.

Jiahao Yao, Marin Bukov, and Lin Lin. Policy gradient based quantum approximate optimization algorithm. In *Mathematical and Scientific Machine Learning*, pages 605–634. PMLR, 2020a.

Jiahao Yao, Paul Köttering, Hans Gundlach, Lin Lin, and Marin Bukov. Noise-robust end-to-end quantum control using deep autoregressive policy networks. *arXiv preprint arXiv:2012.06701*, 2020b.

Jiahao Yao, Lin Lin, and Marin Bukov. Reinforcement learning for many-body ground state preparation based on counter-diabatic driving. *arXiv preprint arXiv:2010.03655*, 2020c. URL https://arxiv.org/abs/2010.03655.

Weirui Ye, Shaohuai Liu, Thanard Kurutach, Pieter Abbeel, and Yang Gao. Mastering atari games with limited data. *arXiv preprint arXiv:2111.00210v2*, Oct 2021. URL http://arxiv.org/abs/2111.00210v2.

Shi-Xin Zhang, Chang-Yu Hsieh, Shengyu Zhang, and Hong Yao. Neural predictor based quantum architecture search. *arXiv preprint arXiv:2103.06524v1*, Mar 2021. URL http://arxiv.org/abs/2103.06524v1.

Shi-Xin Zhang, Chang-Yu Hsieh, Shengyu Zhang, and Hong Yao. Differentiable quantum architecture search. *Quantum Science and Technology*, 7(4):045023, Aug 2022. URL https://doi.org/10.1088/2058-9565/ac87cd.

Hui Zhou, Yunlan Ji, Xinfang Nie, Xiaodong Yang, Xi Chen, Ji Bian, and Xinhua Peng. Experimental realization of shortcuts to adiabaticity in a nonintegrable spin chain by local counterdiabatic driving. *Physical Review Applied*, 13(4):044059, 2020a.

Xiangzhen Zhou, Yuan Feng, and Sanjiang Li. A monte carlo tree search framework for quantum circuit transformation. *arXiv preprint arXiv:2008.09331v2*, Aug 2020b. URL http://arxiv.org/abs/2008.09331v2.

Linghua Zhu, Ho Lun Tang, George S Barron, Nicholas J Mayhall, Edwin Barnes, and Sophia E Economou. An adaptive quantum approximate optimization algorithm for solving combinatorial problems on a quantum computer. *arXiv preprint arXiv:2005.10258*, 2020. URL https://arxiv.org/abs/2005.10258.

Barret Zoph and Quoc V. Le. Neural architecture search with reinforcement learning. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=r1Ue8Hcxg.

Appendix A. Related works

Hybrid optimization: The generalized QAOA ansatz introduces a discrete and continuous control problem: the discrete degrees of freedom are the gates/unitaries that define the control protocol, while the continuous degrees of freedom are the gate duration. Most reinforcement learning algorithms (Lillicrap et al., 2016; Bertsekas, 2019; Trabucco et al., 2021) typically deal with the control of either discrete or continuous degree of freedom, and hardly consider the discrete and continuous control simultaneously in the policy. Even though the continuous control can always be discretized, it is always beneficial and desirable to consider discrete and continuous variables together, without loss of the flexibility of continuous control. Furthermore, the idea of continuous and discrete optimization can be quite general, and shows up in real world application like robotics (Neunert et al., 2020; Delalleau et al., 2019) and strategic games (Vinyals et al., 2019). Combining the discrete and continuous control together, the control capability of the algorithm can be quickly enhanced. In general, the discrete variables are usually chosen as the categories of actions, and the continuous variables will are naturally given by the strength for each specific action. Our work aims to shed light on the hybrid control in the field of quantum control, and we also hope it will accelerate the research of hybrid discrete-continuous optimization algorithms in the wider community.

Counter diabatic driving: Counter diabatic driving (Sels and Polkovnikov, 2017; Hegade et al., 2022), an example of a shortcut to adiabaticity (STA), introduces an extra auxiliary counter-diabatic (CD) Hamiltonian to suppress transitions (or excitations) between instantaneous eigenvalues.

For a given quantum state $|\psi\rangle$ evolving under a time dependent Hamiltonian $H_0(\lambda(t))$, the Schrödinger equation reads as

$$\begin{aligned} i\hbar\partial_t |\psi\rangle &= H_0(\lambda(t)) |\psi\rangle , \\ |\psi_i\rangle = |\psi_{\mathrm{GS}}(\lambda=0)\rangle , |\psi_*\rangle &= |\psi_{\mathrm{GS}}(\lambda=1)\rangle . \end{aligned} \tag{A.1}$$

In the rotating frame, Hamiltonian remains stationary under the unitary transformation $U(\lambda(t))$, i.e. in the instantaneous eigenbasis of Hamiltonian $H_0(\lambda)$. The wave function $|\tilde{\psi}\rangle = U(\lambda)|\psi\rangle$ in the rotating frame satisfies the following Schrödinger equation:

$$i\hbar\partial_t|\tilde{\psi}\rangle = \left(\tilde{H}_0(\lambda(t)) - \dot{\lambda}\tilde{\mathcal{A}}_\lambda\right)|\tilde{\psi}\rangle, \tag{A.2}$$

where $\tilde{H}_0(\lambda(t)) = U^\dagger H_0(\lambda(t))U, \tilde{\mathcal{A}}_\lambda = iU^\dagger\partial_\lambda U$. Specifically, instead of being diagonalized, the original Hamiltonian picks up an extra contribution due to the change in the parameter $\lambda(t)$, and the effective Hamiltonian becomes

$$H_0^{\mathrm{eff}} = \tilde{H}_0 - \dot{\lambda}\tilde{\mathcal{A}}_\lambda. \tag{A.3}$$

The idea of the CD driving is to evolve the system with the counterdiabatic Hamiltonian

$$H_{\mathrm{CD}}(t) = H_0 + \dot{\lambda}\mathcal{A}_\lambda. \tag{A.4}$$

Importantly, in the moving frame $H_{\mathrm{CD}}^{\mathrm{eff}}(t) = \tilde{H}_0$ is *stationary and no transitions occur*.

However, in practice, the precise counter-diabatic Hamiltonian is intractable and usually approximated by different methods. A good number of prior works (Passarelli et al., 2020; Hartmann and Lechner, 2019; Hegade et al., 2022, 2021a; Zhou et al., 2020a; Wurtz and Love, 2021; Hegade

et al., 2021b) are based on the concept of a variational approximation to the CD Hamiltonian (Sels and Polkovnikov, 2017). Most of these works typically make use of an analytically computed expression available for few-qubit systems; they first derive the continuous form of the variational gauge potential, and then discretize the underlying dynamics using the Trotter-Suzuki formula In this work, we aim to bypass these constraints by applying the variational generalized QAOA ansatz using additional gates, generated by terms that occur in the approximation to the variational adiabatic gauge potential. These extra gates can provide a shortcut to the preparation of the ground state, compared to the original alternating QAOA ansatz. Physically, this shortcut results in shorter circuit simulation times, which provices a significant advantage on noisy NISQ devices.

AutoML and neural architecture search: Automatic machine learning or AutoML has recently attracted lots of attentions as it reduces human efforts in designing the neural architecture from experience and instead leverage the computational power to search the best configuration. One of the most pronounced examples are neural architecture search (NAS) and their variants (Zoph and Le, 2017; Elsken et al., 2018; Liu et al., 2019; Cai et al., 2018; Real et al., 2020), where reinforcement learning or evolutionary strategies are used to find a better network architecture. Inspired by the success of AutoML, the architecture of quantum circuits can also be improved by machine learning algorithms, such as the quantum version of Neural Architecture Search (Wang et al., 2021a,b; Zhang et al., 2021, 2022; Kuo et al., 2021). These prior works interpret the problem as quantum compiling problems, which assembles quantum gates in the low level. Instead of exposing a huge number of choice alternatives for the search algorithms, our work specially uses the variational gauge potentials as the Hamiltonian pool for the search algorithm in a computation-efficient way. Compared with QAOA, MCTS-QAOA has more degree of freedom to approximate the unitary operator; compared with the quantum compiling, it does not search gates in the low level due to the constraint of computations. From this perspective, our method hits the sweet spot between the expressivity and efficiency.

Appendix B. Setup of physical models

We first give a brief review on the physical models used in the numerical experiments. In all experiments, we choose the target state as the ground state of the Hamiltonian H, denoted $|\psi_{\mathrm{GS}}(H)\rangle$. The spin-$1/2$ matrices describing spin i are denoted by X_i, Y_i, Z_i. In contrast to the models considered in (Yao et al., 2020b,c), the Hamiltonians used in this work is normalized by its operator norm $\|H\|$, i.e., we use $H/\|H\|$ instead of the original Hamiltonian H. The reason for introducing the normalized Hamiltonian is as follows. For generic Hamiltonians H (e.g., sparse matrices), the cost of performing a Hamiltonian evolution $e^{-iH\alpha}$ on a quantum device is $\Omega(\|H\|\alpha)$ (Berry et al., 2007; Low and Chuang, 2017). Due to the potential differences between the Hamiltonian norms in the Hamiltonian pool $\mathcal{A}$, using a normalized Hamiltonian $H/\|H\|$ (the corresponding duration parameter α is thus multiplied by $\|H\|$) can lead to a more realistic estimate of the cost of the quantum simulation. Due to this multiplication factor, the duration shown in the results below is larger than that presented in (Yao et al., 2020b,c).

One-dimensional (1D) Ising model

The spin-$1/2$ Ising Hamiltonian reads as:

$$H \;=\; H_1+H_2, \quad H_1=\sum_{i=1}^{N} JZ_{i+1}Z_i+h_zZ_i, \quad H_2=\sum_{i=1}^{N} h_xX_i,$$

where N is the number of qubits and the parameters are set as $h_z/J=0.4523$ and $h_x/J=0.4045$ (Kim and Huse, 2013). These parameters are close to the critical line of the model in the thermodynamic limit, where the quantum phase transition occurs. They are also reported in Ref. (Matos et al., 2021) to be in the most challenging parameter region using QAOA. We use periodic boundary conditions here. The initial state for this experiment is given by z-polarized product state, i.e. $|\psi_{\text{init}}\rangle = |\uparrow\cdots\uparrow\rangle$.

For the Hamiltonian pool, we use $\mathcal{A}=\left\{J\frac{H_1}{||H_1||}, J\frac{H_2}{||H_2||}, J\frac{A_1}{||A_1||}, J\frac{A_2}{||A_2||}, J\frac{A_3}{||A_3||}\right\}$, where $A_1=\sum_{i=1}^{N} Y_i,\ A_2=\sum_{i=1}^{N} X_iY_i+Y_iX_i,\ A_3=\sum_{i=1}^{N} Z_iY_i+Y_iZ_i$. The operators A_j are precisely the first three terms in the expansion for the adiabatic gauge potential of the translation-invariant 1D Ising model (Yao et al., 2020c).

Two-dimensional (2D) Ising model

The 2D spin-$1/2$ transverse-field Ising model reads:

$$H=H_1+H_2, \quad H_1=J\sum_{\langle i,j\rangle} Z_iZ_j+h_z\sum_j Z_j,\ H_2=\sum_j h_xX_j,$$

where $\langle i,j\rangle$ denotes nearest neighbors on the square lattice. The model parameters are set as $h_z/J=2$ and $h_x/J=3$. The initial state is $|\psi_{\text{init}}\rangle=|\uparrow\rangle$, i.e. z-polarized product state on 2D lattice.

For the Hamiltonian pool, we use $\mathcal{A}=\left\{J\frac{H_1}{||H_1||}, J\frac{H_2}{||H_2||}, J\frac{A_1}{||A_1||}, J\frac{A_2}{||A_2||}, J\frac{A_3}{||A_3||}\right\}$, where $A_1=\sum_j Y_j,\ A_2=\sum_{\langle i,j\rangle} X_iY_j+Y_iX_j,\ A_3=\sum_{\langle i,j\rangle} Z_iY_j+Y_iZ_j$.

Lipkin-Meshkov-Glick (LMG) model

The Lipkin-Meshkov-Glick (LMG) model (Lipkin et al., 1965) reads:

$$H=H_1+H_2, \quad H_1=-\frac{J}{N}\sum_{i,j=1}^{N} X_iX_j,\ H_2=h\sum_{j=1}^{N}\left(Z_j+\frac{1}{2}\right),$$

where J is the interactions trength, and h stands for the magnetic field strength. The LMG model preserves the total spin, and the ground state is contained in an $N+1$ dimensional subspace due to this symmetry. This makes the LMG model particularly interesting because it allows us to simulate its dynamics for a large number of spins, where many-body effects, such as collective phenomena, dominate the physics of the system.

For instance, in the thermodynamic limit $N\to\infty$, the LMG model exhibits a quantum phase transition at $h_c/J=1$ (Botet and Jullien, 1983). The transition is between a ferromagnetic (FM) order in the ground state in the x-direction ($h/J\ll 1$), and the paramagnetic order ($h/J\gg 1$).

For the Hamiltonian pool, we use $\mathcal{A}=\left\{J\frac{H_1}{||H_1||}, J\frac{H_2}{||H_2||}, J\frac{A_1}{||A_1||}, J\frac{A_2}{||A_2||}, J\frac{A_3}{||A_3||}\right\}$, where

$$\begin{aligned} A_1 &= \sum_{j=1}^{N} Y_j, \\ A_2 &= \frac{1}{N}\left(\sum_{j=1}^{N} Y_j\right)\left(\sum_{j=1}^{N} X_j\right) + \frac{1}{N}\left(\sum_{j=1}^{N} X_j\right)\left(\sum_{j=1}^{N} Y_j\right), \\ A_3 &= \frac{1}{N}\left(\sum_{j=1}^{N} Y_j\right)\left(\sum_{j=1}^{N}\left(Z_j+\frac{1}{2}\right)\right) + \frac{1}{N}\left(\sum_{j=1}^{N}\left(Z_j+\frac{1}{2}\right)\right)\left(\sum_{j=1}^{N} Y_j\right). \end{aligned} \tag{B.1}$$

Appendix C. Noise models

An essential part of our study is the performance of the algorithms in the presence of noise. As mentioned in the main text, noise sets the current bottle neck for reliable quantum computation. Therefore, it is of primary importance for the near-term utility of quantum computers to develop stable and noise-robust manipulation algorithms.

We use the following three noise models in our numerical experiments: (i) classical measurement noise, (ii) quantum measurement error which micmic the situation on present-day NISQ devices, and (iii) gate rotation error noise.

Classical measurement Gaussian noise is added to the cost function according to

$$\mathcal{L}_\gamma(\boldsymbol{\theta}) = \mathcal{L}(\boldsymbol{\theta}) + \epsilon_\gamma,$$

where $\epsilon_\gamma \sim \mathcal{N}(0,\gamma^2)$ and γ denotes the noise strength, and $\mathcal{N}$ is the normal distribution. Gaussian noise models various kinds of uncertainty present in experiments using an additive Gaussian random variable, which follows from the Central Limit theorem.

Quantum measurement noise:

$$\mathcal{L}_Q(\boldsymbol{\theta}) = \mathcal{L}(\boldsymbol{\theta}) + \epsilon_Q,$$

where the noise strength depends on the strength of the energy quantum fluctuations

$$\Delta\mathcal{E} = N^{-1}\sqrt{\langle\psi(T)|H^2|\psi(T)\rangle - \langle\psi(T)|H|\psi(T)\rangle^2},$$

and ϵ_Q is randomly sampled from $\mathcal{N}(0,\Delta\mathcal{E}^2)$. Quantum noise models the uncertainty arising from quantum measurements. For instance, quantum fluctuations are large when the evolved quantum state is far away from the target, while they decrease when the final state approaches the target ground state.

Gate rotation error noise:

$$\mathcal{L}_\delta(\boldsymbol{\theta}) = \mathcal{L}(\boldsymbol{\theta}'), \quad \boldsymbol{\theta}' = (\{\alpha_i + \alpha_i\epsilon_i\}_{i=1}^{q}, \boldsymbol{\tau})$$

where gate error strengths are multiplicative and the corresponding ratios are $\epsilon_i \sim \mathcal{N}(0,\delta^2)$ for some simulation parameter δ which controls the noise strength. Gate rotation errors (Sung et al., 2020) present yet another common noise source, which arises due to imperfections or lack of calibration in the quantum computer hardware.

Appendix D. Details for the natural policy gradient with entropy regularization

For a general d-dimensional Gaussian distribution $\mathcal{N}(\mu, \Sigma)$, the Shannon entropy is defined as $\mathbb{E}(-\log(p(x)))$, where $p(x) = (2\pi)^{-\frac{d}{2}}|\Sigma|^{-\frac{1}{2}}\exp\big(-\frac{1}{2}(x-\mu)^\top\Sigma^{-1}(x-\mu)\big)$. Hence

$$\begin{aligned}
\mathbb{E}(-\log(p(x))) &= -\mathbb{E}\log\left[(2\pi)^{-\frac{d}{2}}|\Sigma|^{-\frac{1}{2}}\exp\left(-\frac{1}{2}(x-\mu)^\top\Sigma^{-1}(x-\mu)\right)\right] \\
&= \mathbb{E}\left[\frac{d}{2}\log 2\pi + \frac{1}{2}\log|\Sigma| + \frac{1}{2}(x-\mu)^\top\Sigma^{-1}(x-\mu)\right] \\
&= \frac{d}{2}\log 2\pi + \frac{1}{2}\log|\Sigma| + \frac{1}{2}\mathbb{E}(x-\mu)^\top\Sigma^{-1}(x-\mu) \\
&= \frac{d}{2}\log 2\pi + \frac{1}{2}\log|\Sigma| + \frac{1}{2}\mathbb{E}\,\mathrm{Tr}\Big((x-\mu)^\top\Sigma^{-1}(x-\mu)\Big) \\
&= \frac{d}{2}\log 2\pi + \frac{1}{2}\log|\Sigma| + \frac{1}{2}\mathbb{E}\,\mathrm{Tr}\Big(\Sigma^{-1}(x-\mu)(x-\mu)^\top\Big) \\
&= \frac{d}{2}\log 2\pi + \frac{1}{2}\log|\Sigma| + \frac{d}{2}.
\end{aligned}$$

Omitting the constants, it is equivalent to take the entropy as $\frac{1}{2}\log|\Sigma|$. For the model used, the probability distribution is a product of normal distribution, i.e., Σ is a diagonal matrix with length q and diagonal elements σ_i, so the corresponding entropy function is $\mathbb{E}(-\log(p(x))) = \sum_{i=1}^q \log\sigma_i$.

In the implementation, we adopt the parameterization $\sigma_i = \exp(t_i)$ to assure that σ_i is positive. Then for the distribution $\mathcal{N}(\mu_i, \sigma_i)$, we have

$$\log p_i(x) = -\frac{(x-\mu_i)^2}{2\sigma_i^2} - \log\sigma_i - \frac{1}{2}\log(2\pi) = -\frac{1}{2}(x-\mu_i)^2e^{-2t_i} - t_i - \frac{1}{2}\log(2\pi),$$

and

$$\nabla\log p_i(x) = ((x-\mu_i)e^{-2t_i}, (x-\mu_i)^2e^{-2t_i} - 1)^\top,$$

where the gradient is taken with respect to the parameters. Since $\{\delta_i\}_{i=1}^q$ are independent, the Fisher information matrix is a block diagonal matrix with the i-th block equal to

$$F_i = \mathbb{E}\nabla\log p_i(x)\nabla\log p_i(x)^\top = \mathbb{E}\begin{bmatrix} \frac{(x-\mu_i)^2}{\sigma_i^4} & \frac{(x-\mu_i)^3}{\sigma_i^3} - \frac{(x-\mu_i)}{\sigma_i} \\ \frac{(x-\mu_i)^3}{\sigma_i^3} - \frac{(x-\mu_i)}{\sigma_i} & \frac{(x-\mu_i)^4}{\sigma_i^4} - \frac{2(x-\mu_i)^2}{\sigma_i^2} + 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sigma_i^2} & 0 \\ 0 & 2 \end{bmatrix}.$$

Recall that for a fixed gate sequence $\boldsymbol{\tau}$, we set $R(\boldsymbol{\delta}) = -E\left(\left\{\frac{T\mathfrak{g}(\delta_j)}{\sum_k\mathfrak{g}(\delta_k)}\right\}_{j=1}^q, \boldsymbol{\tau}\right)/N$, where $\mathfrak{g}$ denotes the sigmoid function, and

$$\mathcal{J}(\{\mu_j,\sigma_j\}_{j=1}^q) = \mathbb{E}_{\delta_j\sim\mathcal{N}(\mu_j,\sigma_j)}R(\boldsymbol{\delta}) + \beta_S^{-1}\sum_{j=1}^q\log\sigma_j.$$

Hence the gradient of $\mathcal{J}$ is

$$\mathbb{E}R(\boldsymbol{\delta})\nabla\log p(\boldsymbol{\delta}) + \beta_S^{-1}\nabla\sum_{j=1}^q\log\sigma_j,$$

where the gradient is taken with respect to the parameters, and $p(\boldsymbol{\delta}) = \prod_{i=1}^{q} p_i(\delta_i)$. Therefore, the unbiased estimators for the variables are

$$\begin{bmatrix} \frac{\partial \mathcal{J}}{\partial \mu_j} \\ \frac{\partial \mathcal{J}}{\partial t_j} \end{bmatrix} \leftarrow \begin{bmatrix} R(\boldsymbol{\delta})\xi_j/\sigma_j \\ R(\boldsymbol{\delta})(\xi_j^2 - 1) + \beta_S^{-1} \end{bmatrix},$$

where ξ_j are independent standard normal variables and $\delta_j = \sigma_j \xi_j + \mu_j$. As a result, the unbiased estimators for the natural gradient direction become

$$F_j^{-1} \begin{bmatrix} \frac{\partial \mathcal{J}}{\partial \mu_j} \\ \frac{\partial \mathcal{J}}{\partial t_j} \end{bmatrix} \leftarrow \begin{bmatrix} \sigma_j R(\boldsymbol{\delta})\xi_j \\ \frac{1}{2} R(\boldsymbol{\delta})(\xi_j^2 - 1) + \frac{1}{2}\beta_S^{-1} \end{bmatrix},$$

since F is a block diagonal matrix with the j-th block given by F_j.

Appendix E. Additional experiment results

In Section 5.1, we have presented a comparison between the RL-QAOA method and MCTS-QAOA for three different physics models with the quantum noise. In this section, we report the test results with the other types of noise, namely the results with the Gaussian noise, the results with the gate rotation error, and the results when no noise is considered (cf. Appendix C). We can observe from the comparison that MCTS-QAOA's performance is much more stable and accurate.

From Figure 6, one can observe similar behavior the two methods as in Section 5.1, i.e., MCTS-QAOA outperforms RL-QAOA in all settings and the gaps grow larger in the regime of large total gate durations. The raw data for the energy ratio obtained by MCTS-QAOA is summarized in Table 2 (highlighted in bold), which offers a more visually and quantitatively convenient comparison across different models.

Appendix F. Additional numerical results on the energy landscape

In Section 5.2 we reported the discrete landscape of the generalized QAOA ansatz under the condition that the continuous variables are solved with high quality with the improved NPG solver. Here we include the landscape under another physical model, i.e. 2D Ising model. We consider the case where $(|\mathcal{A}|, q) = (5, 8)$, and the total number of gate sequences is thus 81920. Similar to the plots displayed in Section 5.2, the landscape with a longer total duration ($JT = 50$) features a dominant cluster at the rightmost part of the histogram. When the total duration is smaller, the number of clusters increases, and is shifted to the left.

Figure 8 shows the influence of the parameter h/J in the discrete landscape for the LMG model with gate duration $JT = 1500$ and $N = 100$. When $h/J = 0.8$ and $h/J = 0.99$, the rightmost peak in the energy ratio histogram gets close to 1, which means that reaching the ground state would be a easy task in these two cases. The more difficult cases lies in between, for example when $h/J = 0.95$. For the parameter $h/J = 0.9$ we choose in the main text, there is a bigger gap (cf. Fig. 4(c)) between the rightmost peak of the energy ratio and 1, which means the problem we choose to solve is relatively challenging.

JT	Gate rotation noise	Quantum noise	Gaussian noise	No noise
	(E/E_{GS})			
(*Model*)	(a) **Ising 1D**			
10.0	-0.0208 (-0.0208)	-0.0219 (-0.0238)	-0.0225 (-0.0209)	-0.0210 (-0.0207)
20.0	0.4907 (0.4884)	0.4862 (0.4863)	0.4903 (0.4905)	0.4907 (0.4908)
30.0	0.7849 (0.7844)	0.7830 (0.7796)	0.7825 (0.7833)	0.7850 (0.7850)
40.0	0.9481 (0.9486)	0.9521 (0.9477)	0.9512 (0.9513)	0.9516 (0.9527)
50.0	0.9503 (0.9499)	0.9499 (0.9564)	0.9505 (0.9581)	0.9559 (0.9574)
60.0	0.9489 (0.9614)	0.9526 (0.9540)	0.9576 (0.9560)	0.9570 (0.9621)
120.0	0.9424 (0.9495)	0.9543 (0.9548)	0.9606 (0.9524)	0.9548 (0.9602)
180.0	0.9495 (0.9415)	0.9486 (0.9502)	0.9582 (0.9556)	0.9514 (0.9543)
(*Model*)	(b) **Ising 2D**			
10.0	-0.1586 (-0.1586)	-0.1645 (-0.1610)	-0.1589 (-0.1614)	-0.1587 (-0.1587)
20.0	0.2688 (0.2692)	0.2663 (0.2672)	0.2680 (0.2677)	0.2730 (0.2688)
30.0	0.7771 (0.7777)	0.7786 (0.7800)	0.7799 (0.7797)	0.7812 (0.7812)
40.0	0.9635 (0.9635)	0.9641 (0.9651)	0.9647 (0.9633)	0.9654 (0.9635)
50.0	0.9984 (0.9984)	0.9979 (0.9982)	0.9982 (0.9983)	0.9985 (0.9985)
60.0	0.9980 (0.9979)	0.9978 (0.9981)	0.9981 (0.9965)	0.9986 (0.9984)
(*Model*)	(c) **LMG**			
30.0	**0.4775** (0.4774)	**0.4762** (0.4729)	**0.4766** (0.4770)	**0.4776** (0.4774)
60.0	**0.5828** (0.5828)	**0.5792** (0.5803)	**0.5818** (0.5815)	**0.5828** (0.5828)
100.0	**0.7471** (0.7467)	**0.7447** (0.7468)	**0.7459** (0.7460)	**0.7472** (0.7471)
200.0	**0.8591** (0.8592)	**0.8561** (0.8568)	**0.8583** (0.8587)	**0.8591** (0.8591)
300.0	**0.9093** (0.9098)	**0.9091** (0.9077)	**0.9095** (0.9093)	**0.9101** (0.9104)
500.0	**0.9312** (0.9368)	**0.9349** (0.9363)	**0.9367** (0.9367)	**0.9372** (0.9371)
1000.0	**0.9463** (0.9484)	**0.9522** (0.9475)	**0.9505** (0.9510)	**0.9518** (0.9523)
1500.0	**0.9474** (0.9436)	**0.9385** (0.9336)	**0.9444** (0.9524)	**0.9499** (0.9453)
2000.0	**0.9447** (0.9448)	**0.9646** (0.9496)	**0.9648** (0.9521)	**0.9636** (0.9562)

Table 2: **Energy ratio obtained by MCTS-QAOA**: MCTS-QAOA using the Hamiltonian pool without identity (**bold**, see Appendix B) and with identity operator (gray in the parenthesis, see Appendix G). Sector (a): 1D spin-$1/2$ Ising chain ($N = 8$); Sector (b): 2D spin-$1/2$ Ising chain ($N = 3 \times 3$); Sector (c): LMG model ($N = 100$) at $h/J = 0.9$. We use $\gamma = 0.1$ for Gaussian noise and $\delta = 0.1$ for the gate rotation noise (see Appendix. C).

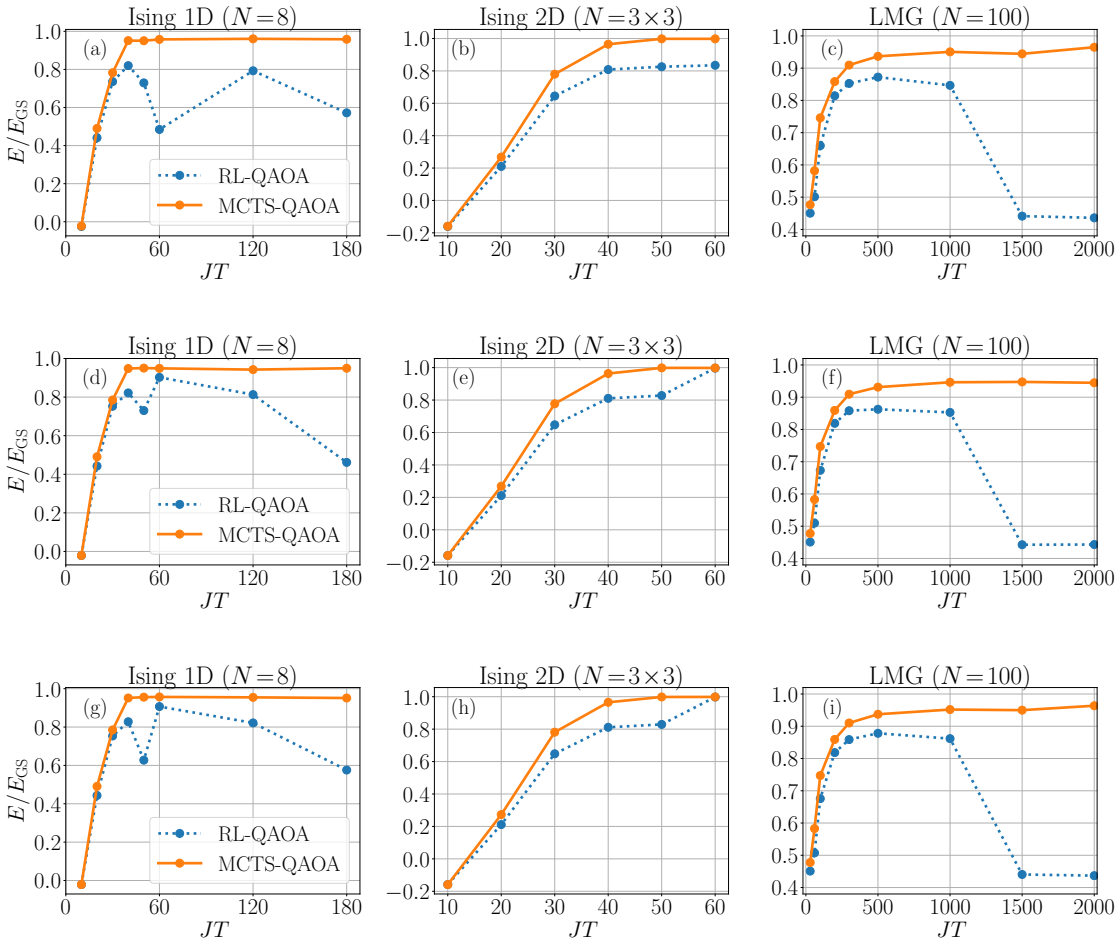

Figure 6: (**experiment with other types of noise models or without noise**) comparison between MCTS-QAOA and RL-QAOA. The physics setup is the same as that in Figure 2. (a-c): Gaussian noise with $\gamma = 0.1$; (d-f): gate rotation noise with $\delta = 0.1$; (g-i): experiments without noise (cf. Appendix C).

Appendix G. Physical models with the identity action

The generalized QAOA ansatz provides us the freedom of adding different Hamiltonians to the Hamiltonian pool. One meaningful addition is the identity operator. Here, the identity operator corresponds to the identity gate that does not move the quantum state. If we take $\tilde{H} = \mathbf{0}$, then its corresponding unitary gate will be identity, i.e. $\exp\left(-i\tilde{H}\tilde{\alpha}\right) = \mathbf{I}$. This approach adds an extra amount of freedom to the optimization since the quantum control no longer needs to figure out how to *exactly* distribute the gate duration budgets to different gates so as to reach the ground state. In

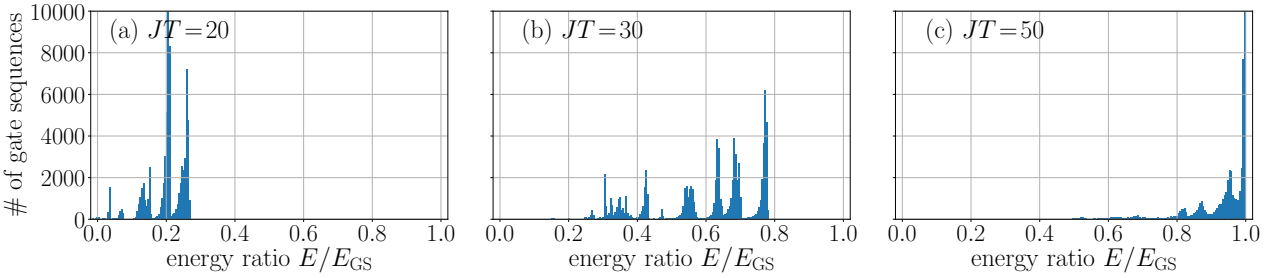

Figure 7: **Discrete landscape of 2D Ising model**: (a-c): Histograms of the energy ratio optimized by the improved natural gradient solver for $JT = 20, 30, 50$, respectively. $N_{\mathrm{hist}} = 81920$ samples are chosen from the discrete gate sequences of generalized QAOA with parameters $q = 8$ and $|\mathcal{A}| = 5$.

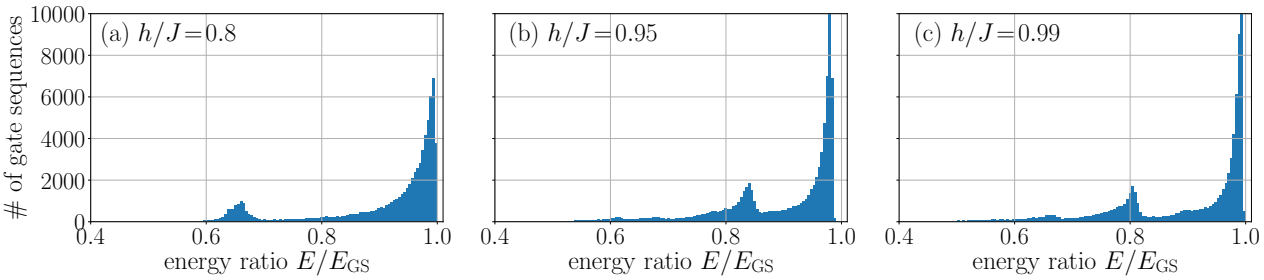

Figure 8: **Discrete landscape of LMG model with respect to different parameter** h/J: (a-c): Histograms of the energy ratio optimized by the improved natural gradient solver for $h/J = 0.8, 0.95$, and 0.99, respectively with gate duration $JT = 1500$. $N_{\mathrm{hist}} = 81920$ samples are chosen from the discrete gate sequences of generalized QAOA with parameters $q = 8$, $|\mathcal{A}| = 5$ and $N = 100$. For the LMG model, the gap between the right-most peak and 1 is larger when h/J is between 0.8 and 0.99.

other words, the original optimization problem (see Eqn. 4.2) becomes the relaxed form:

$$\min_{\{\alpha_j\}_{j=1}^q} \left\{ E(\{\alpha_j\}_{j=1}^q, \boldsymbol{\tau}) : \sum_{j=1}^{q} \alpha_j \leq T;\ 0 \leq \alpha_j \leq T \right\}. \tag{G.1}$$

With the identity action, the extended action space becomes $\mathcal{A} = \left\{ \mathbf{0}, \frac{H_1}{||H_1||}, \frac{H_2}{||H_2||}, \frac{A_1}{||A_1||}, \frac{A_2}{||A_2||}, \frac{A_3}{||A_3||} \right\}$, with the definitions shown in Appendix. B for three different physics models.

In this setting, a similar behavior is observed as in Figure 2 and Figure 6, which is shown in Figure 9. We conclude that MCTS-QAOA outperforms RL-QAOA in all settings and MCTS-QAOA still maintains a robust performance when that of RL-QAOA begins to deteriorate in the regime of large total gate durations. The raw data of the energy ratio obtained by MCTS-QAOA is reported in Table 2 (highlighted in gray), which also gives a direct comparison with the energy ratios obtained without the identity action. It can be seen that the performance of MCTS-QAOA in this setting is on par with the setting presented in the main text.

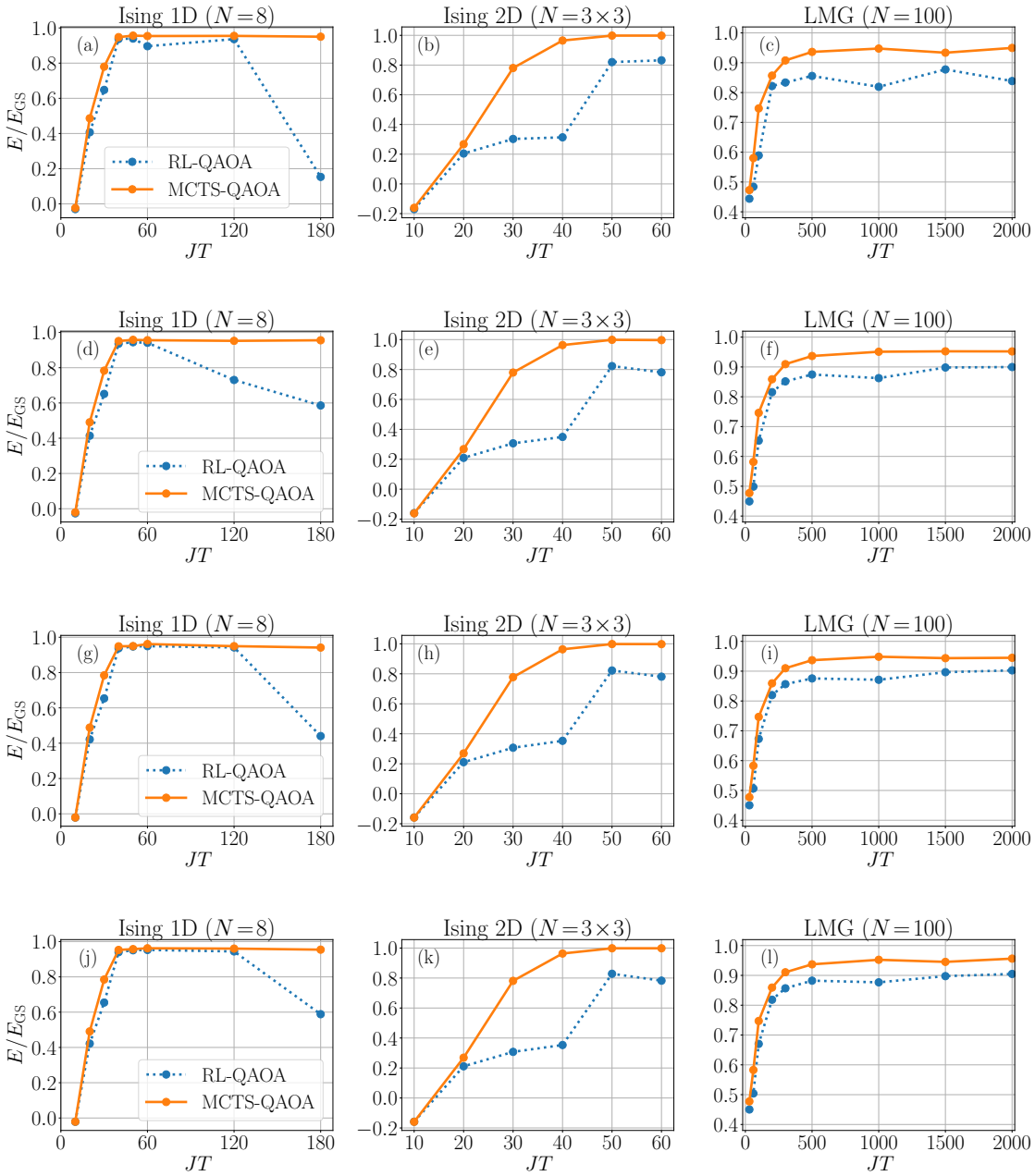

Figure 9: Comparison between MCTS-QAOA and RL-QAOA using the Hamiltonian pool with the identity operation. The physics setup is the same as that in Figure 2. (a-c): quantum measurement noise; (d-f): Gaussian noise with $\gamma = 0.1$; (g-i): gate rotation noise with $\delta = 0.1$; (k-l): experiments without noise (cf. Appendix C).