

## Adaptive quantum approximate optimization algorithm for solving combinatorial problems on a quantum computer

Linghua Zhu <sup>1,\*</sup> Ho Lun Tang,<sup>1</sup> George S. Barron <sup>1</sup> F. A. Calderon-Vargas,<sup>1</sup>

Nicholas J. Mayhall,<sup>2</sup> Edwin Barnes,<sup>1</sup> and Sophia E. Economou<sup>1,†</sup>

<sup>1</sup>*Department of Physics, Virginia Tech, Blacksburg, Virginia 24061, USA*

<sup>2</sup>*Department of Chemistry, Virginia Tech, Blacksburg, Virginia 24061, USA*



(Received 25 May 2020; revised 22 December 2020; accepted 17 June 2022; published 11 July 2022)

The quantum approximate optimization algorithm (QAOA) is a hybrid variational quantum-classical algorithm that solves combinatorial optimization problems. While there is evidence suggesting that the fixed form of the standard QAOA *Ansatz* is not optimal, there is no systematic approach for finding better *Ansätze*. We address this problem by developing an iterative version of QAOA that is problem tailored, and which can also be adapted to specific hardware constraints. We simulate the algorithm on a class of Max-Cut graph problems and show that it converges much faster than the standard QAOA, while simultaneously reducing the required number of CNOT gates and optimization parameters. We provide evidence that this speedup is connected to the concept of shortcuts to adiabaticity.

DOI: [10.1103/PhysRevResearch.4.033029](https://doi.org/10.1103/PhysRevResearch.4.033029)

### I. INTRODUCTION

Many important computationally hard combinatorial optimization problems such as Max-Cut, graph coloring, traveling salesman, and scheduling management [1–3] can be mapped to Ising Hamiltonians whose ground states provide the solutions. One can in principle solve these optimization problems on a quantum computer by initializing the quantum device in the ground state of a Hamiltonian that is easy to prepare and adiabatically tuning the latter into the problem Hamiltonian. In a digital quantum computer, this translates into a Trotterized version of the adiabatic evolution operator, which is the alternating product of the evolution operators corresponding to the initial mixer and the problem (Ising) Hamiltonians. In the limit of an infinite product, this Trotterized form becomes exact.

QAOA is a hybrid quantum-classical variational algorithm that uses a finite-order version of this evolution operator to prepare wave-function *Ansätze* on a quantum processor [4–7]. QAOA is performed by variationally minimizing the expectation value of the Ising Hamiltonian with respect to the parameters in the *Ansatz*. The quantum processor is also used to measure energy expectation values, while the optimization is done on a classical computer. There has been a lot of progress on QAOA recently on both the experimental and theoretical fronts [1,8–14]. There is evidence suggesting that

QAOA may provide a significant quantum advantage over classical algorithms [14–16], and that it is computationally universal [17,18].

Despite these advances, there are limitations and potential issues with this algorithm. The performance improves with the number of alternating layers in the *Ansatz*, but the latter is limited by coherence times in existing and near-term quantum processors. Moreover, more layers implies more variational parameters, which introduces challenges for the classical optimizer [19]. Furthermore, Ref. [20] points out that the locality and symmetry of QAOA can also severely limit its performance. These issues can be attributed to, or are at least exacerbated by, the form of the QAOA *Ansatz*. In particular, short-depth QAOA is not really the digitized version of the adiabatic problem, but rather an ad hoc *Ansatz*, and as a result should not be expected to perform optimally, or even well. A short-depth *Ansatz* that is further tailored to a given combinatorial problem could therefore address the issues with the standard QAOA *Ansatz*. However, identifying such an alternative is a highly nontrivial problem given the vast space of possible *Ansätze*.

In this work, we propose an iterative version of QAOA termed adaptive derivative assembled problem tailored-quantum approximate optimization algorithm (ADAPT-QAOA). Our algorithm grows the *Ansatz* two operators at a time by using a gradient criterion to systematically select the QAOA mixer from a predefined operator pool. While ADAPT-QAOA is general and can be applied to any optimization problem, we focus on Max-Cut to quantify its performance. We find that when entangling gates are included in the operator pool, there is a dramatically faster convergence compared to standard QAOA. Surprisingly, despite the introduction of entangling gates as mixers, these improvements come with a reduction in the numbers of both the optimization parameters and the CNOT gates by approximately 50% each compared to

\*zlinghua18@vt.edu

†economou@vt.edu

standard QAOA. We summarize the details of our algorithm and its performance in Sec. II and provide evidence that this improved performance is related to the concept of shortcuts to adiabaticity [21–24] in Sec. III. Finally, we conclude in Sec. IV with a look ahead.

## II. ADAPT-QAOA

### A. Framework

In QAOA [4,5], the variational *Ansatz* consists of  $p$  layers, each containing the cost Hamiltonian  $H_C$  and a mixer,  $H_M$ :

$$|\psi_p(\vec{\gamma}, \vec{\beta})\rangle = \left( \prod_{k=1}^p [e^{-iH_M\beta_k} e^{-iH_C\gamma_k}] \right) |\psi_{\text{ref}}\rangle, \quad (1)$$

where  $|\psi_{\text{ref}}\rangle = |+\rangle^{\otimes n}$ ,  $n$  is the number of qubits, and  $\vec{\gamma}$  and  $\vec{\beta}$  are sets of variational parameters. If these parameters are chosen such that  $\langle \psi_p(\vec{\gamma}, \vec{\beta}) | H_C | \psi_p(\vec{\gamma}, \vec{\beta}) \rangle$  is minimized, then the resulting energy and state provide an approximate solution to the optimization problem encoded in  $H_C$ . The accuracy of the result and the efficiency with which it can be obtained depend sensitively on  $H_M$ . In the standard QAOA *Ansatz*, the mixer is chosen to be a single-qubit  $X$  rotation applied to all qubits. A few papers have suggested modifications to the standard QAOA *Ansatz* for specific problems and hardware architectures [25–27]. These interesting results reveal the potential advantages of the QAOA *Ansatz* but do not provide a universal strategy for choosing mixers that works across a broad range of optimization problems.

In this work, we replace the single, fixed mixer  $H_M$  by a set of mixers  $A_k$  that change from one layer to the next:

$$|\psi_p(\vec{\gamma}, \vec{\beta})\rangle = \left( \prod_{k=1}^p [e^{-iA_k\beta_k} e^{-iH_C\gamma_k}] \right) |\psi_{\text{ref}}\rangle. \quad (2)$$

We build up this *Ansatz* iteratively, one layer at a time, in a way that is determined by  $H_C$ . This iterative process is inspired by the variational quantum eigensolver algorithm, ADAPT-VQE [28,29]. It can be summarized by three basic steps: First, define the operator set  $\{A_j\}$  (called the mixer pool, and where  $A_j = A_j^\dagger$ ) and select a suitable reference state to be the initial state:  $|\psi^{(0)}\rangle = |\psi_{\text{ref}}\rangle$ . Here, we choose  $|\psi_{\text{ref}}\rangle = |+\rangle^{\otimes n}$  as in the standard QAOA. We will return shortly to the question of how to choose the pool. Second, prepare the current *Ansatz*  $|\psi^{(k-1)}\rangle$  on the quantum processor and measure the energy gradient with respect to the pool, the  $j$ th component of which is given by  $-i \langle \psi^{(k-1)} | e^{iH_C\gamma_k} [H_C, A_j] e^{-iH_C\gamma_k} | \psi^{(k-1)} \rangle$ , where the new variational parameter  $\gamma_k$  is set to a predefined value  $\gamma_0$ . For the measurement, we can decompose the commutator into linear combinations of Pauli strings and measure the expectation values of the strings using general variational quantum algorithm methods [30]. If the norm of the gradient is below a predefined threshold, then the algorithm stops, and the current state and energy estimate approximate the desired solution. If the gradient threshold is not met, modify the *Ansatz* by adding the operator,  $A_{\text{max}}^{(k)}$ , associated with the largest component of the gradient:  $|\psi^{(k)}\rangle = e^{-iA_{\text{max}}^{(k)}\beta_k} e^{-iH_C\gamma_k} |\psi^{(k-1)}\rangle$ , where  $\beta_k$  is a new variational parameter. Third, optimize all parameters currently in the *Ansatz*,  $\beta_m, \gamma_m, m = 1, \dots, k$ ,

such that  $\langle \psi^{(k)} | H_C | \psi^{(k)} \rangle$  is minimized, and return to the second step. This algorithm, which we call ADAPT-QAOA, lies somewhere between standard QAOA and ADAPT-VQE in the sense that it possesses the alternating-operator structure of QAOA but enjoys additional flexibility by allowing the mixers to vary over the course of the iterative construction.

### B. Operator pool

The first step in running this algorithm is to define the mixer pool. Define  $Q$  to be the set of qubits. The pool corresponding to the standard QAOA contains only one operator,  $P_{\text{QAOA}} = \{\sum_{i \in Q} X_i\}$ . Here, we introduce two qualitatively different pools: one consisting entirely of single-qubit mixers, and one with both single-qubit and multiqubit entangling gates:  $P_{\text{single}} = \cup_{i \in Q} \{X_i, Y_i\} \cup \{\sum_{i \in Q} Y_i\} \cup P_{\text{QAOA}}$ ,  $P_{\text{multi}} = \cup_{i,j \in Q \times Q} \{B_i C_j | B_i, C_j \in \{X, Y, Z\}\} \cup P_{\text{single}}$ . Because  $P_{\text{QAOA}} \subset P_{\text{single}} \subset P_{\text{multi}}$ , we expect that  $P_{\text{multi}}$  will provide the best performance. The QAOA, single-qubit, and multiqubit pools have  $O(1)$ ,  $O(n)$ , and  $O(n^2)$  elements, respectively.

Max-Cut is a classic (NP-hard) quadratic unconstrained binary optimization problem, and it can be used to solve other optimization problems. Thus, it is a useful benchmarking problem for QAOA and has been used as such in prior works [5,7,11]. It is defined as follows: Given a graph  $G = (V, E)$ , with weight  $w_{i,j}$  for edge  $(i, j)$ , find a cut  $S \subseteq V$  such that  $S \cup \bar{S} = V$ , and  $\sum_{i \in S, j \in \bar{S}, i, j \in E} w_{i,j}$  is maximized. This problem can be encoded in the Ising Hamiltonian

$$H_C = -\frac{1}{2} \sum_{i,j} w_{i,j} (I - Z_i Z_j), \quad (3)$$

where the couplings are given by the edge weights. Each classical state (i.e., tensor product of  $Z$  eigenstates) represents a possible cut.  $H_C$  counts the sum of the weights of the edges connecting one subgraph to the other, and its ground state corresponds to the maximum cut.  $H_C$  has a  $Z_2$  symmetry generated by  $F = \otimes_i X_i$ . Only the  $A_j$  that commute with  $F$  have a nonzero gradient (see Appendix A), so we retain only these Pauli strings (which have an even number of  $Y$  or  $Z$  operators) in our mixer pool.

### C. Performance and resource comparison

We use the Max-Cut problem on regular graphs with  $n = 6$  vertices and degrees  $D = 3$  and  $D = 5$  to benchmark the performance of ADAPT-QAOA. For each type of graph, we analyze 20 instances of random edge weights, which are drawn from the uniform distribution  $U(0, 1)$  [31]. We use Nelder-Mead for the optimization of the variational parameters  $\vec{\beta}$  and  $\vec{\gamma}$ . The gradients used to select new operators are sensitive to the initial values for  $\vec{\gamma}$ . It is natural to initialize these parameters at  $\gamma_0 = 0$  to avoid biasing the optimization. However, as we show in Appendix B,  $\gamma_0 = 0$  is a critical point of the cost function [32]. Moreover, the minimum of the energy in the first layer of ADAPT-QAOA never occurs at  $\gamma_0 = 0$ . Therefore, we shift the initial value  $\gamma_0$  slightly away from zero ( $\gamma_0 = 0.01$ ) to avoid these issues.

In Fig. 1 we show the error as a function of the number of *Ansatz* layers for the standard QAOA and for ADAPT-QAOA using single-qubit and multiqubit mixer pools. For

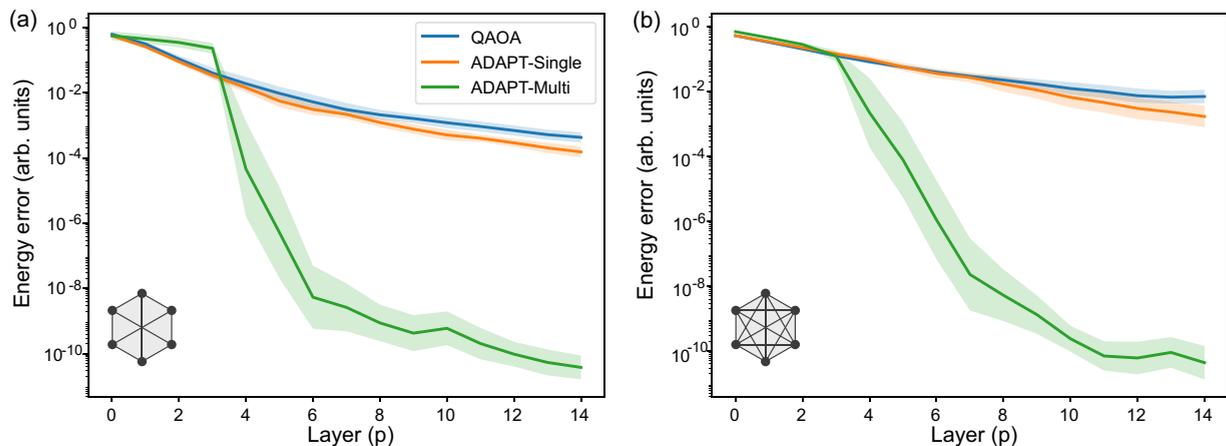


FIG. 1. Comparison of the performance of standard QAOA (blue) with ADAPT-QAOA for the single-qubit (orange) and multiqubit (green) pools. The algorithms are run on the Max-Cut problem for the regular graphs shown in the figure, which have  $n = 6$  vertices and are of degree (a)  $D = 3$  and (b)  $D = 5$ . The energy error (the difference between the energy estimate obtained by the algorithm and the exact ground-state energy of  $H_C$ ) is shown as a function of the number of layers in the *Ansatz*. Results are shown for 20 different instances of edge weights, which are randomly sampled from the uniform distribution  $U(0, 1)$ . The shaded regions indicate 95% confidence intervals.

both three- and five-regular graphs, we find that using the single-qubit mixer pool provides a modest improvement over standard QAOA. On the other hand, the multiqubit pool performs dramatically better, leading to a rapid convergence to the exact solution after only  $\sim 3$  layers. We also find that for the degree-five graphs, standard QAOA and ADAPT-QAOA with single-qubit mixers converge slower than the degree-three case, whereas the performance of ADAPT-QAOA with the multiqubit operator pool remains approximately the same. Note that the particular form of the two-qubit operators in the pool was chosen for its simplicity. In general, one can choose a hardware-tailored operator pool, in the spirit of Ref. [33]. In Appendix C, we show similar results for  $n = 8$  and  $n = 10$  graphs of degree  $D = 2$ , where ADAPT-QAOA with the multiqubit pool substantially outperforms the standard QAOA again. Going to larger values of  $D$  or  $n$  is made challenging by a sharp increase in the number of layers needed to reach convergence, as reported for standard QAOA in Ref. [34].

It is interesting to ask how much the ADAPT-QAOA *Ansätze* differ from the standard QAOA *Ansatz*. We find that when the single-qubit mixer pool is used, the single-qubit operators  $X_i$  are chosen instead of the standard mixer approximately 36.6% of the time for  $n = 6$ ,  $D = 3$  graphs and 25% of the time for  $n = 6$ ,  $D = 5$  graphs. For the multiqubit mixer pool, the algorithm chooses operators other than the standard mixer approximately 75% of the time for  $n = 6$ ,  $D = 3$  graphs and 80% of the time for  $n = 6$ ,  $D = 5$  graphs (see Appendix D). This trend supports the intuitive idea that a more connected graph requires more entanglement for a rapid convergence to the solution.

A crucial question, especially for near-term platforms, is how the different mixer pools compare with respect to resource overhead. Figure 2 shows the number of CNOTs and number of parameters for the three algorithms. The CNOT counts are determined by decomposing each two-qubit operator into two CNOT gates and one or two single-qubit gates. Surprisingly, we find that both the standard QAOA and the single-qubit mixer *Ansätze* in fact have more CNOTs

compared to that constructed from the entangling multiqubit mixer pool. Moreover, on average, the standard QAOA algorithm uses more parameters and CNOTs to reach the same convergence threshold than either version of ADAPT-QAOA. About half as many CNOTs are required for the ADAPT-QAOA multiqubit pool case, despite the fact that the mixers in the multiqubit pool themselves introduce additional CNOT gates on top of those coming from  $H_C$ . Reference [25] proposed using a restricted form of entangling gates in the *Ansatz* to obtain better performance in combinatorial problems at the cost of introducing more variational parameters. In contrast, ADAPT-QAOA provides a systematic way to both improve performance and reduce the number of parameters and CNOTs.

### III. SHORTCUTS TO ADIABATICITY

One may wonder whether there is a physically intuitive way to understand the strikingly better performance of ADAPT-QAOA. Considering that the standard QAOA *Ansatz* has a structure dictated by the adiabatic theorem, a possible explanation is that the ADAPT algorithm is related to shortcuts to adiabaticity (STA). STA, also known as counterdiabatic or transitionless driving, was introduced for quantum systems by Demirplak and Rice [21] and later, independently, by Berry [22,23]. STA has also been explored in the classical context [35,36], including a recent application in biology [37]. The idea is that if we want to drive a system such that it remains in the instantaneous ground state at all times, then by adding a certain term  $\mathcal{H}_{CD}$  to the Hamiltonian, we can achieve this without paying the price of a slow evolution. Although the instantaneous eigenstates of the original Hamiltonian only solve the time-dependent Schrödinger equation in the adiabatic limit, they become exact solutions when the Hamiltonian is updated to include  $\mathcal{H}_{CD}$ . The advantage of STA is that the evolution can be achieved nonadiabatically. Below, we provide evidence that ADAPT-QAOA is indeed related to STA, a likely explanation for why it converges to

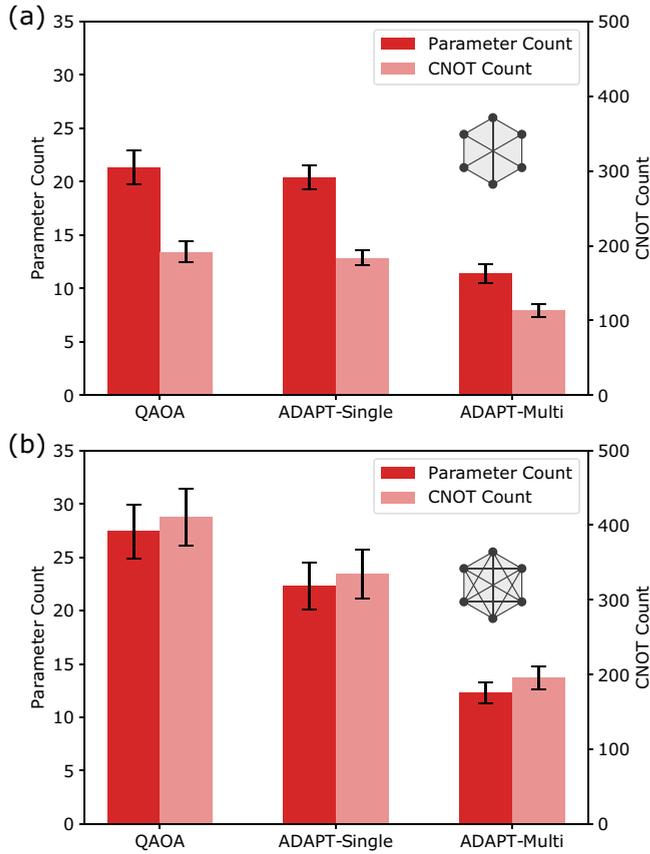


FIG. 2. Resource comparison of the standard QAOA, ADAPT-QAOA with the single-qubit mixer pool, and ADAPT-QAOA with the multiqubit mixer pool for the Max-Cut problem on regular graphs with  $n = 6$  vertices and random edge weights. (a) and (b) show the comparison for graphs of degree  $D = 3$  and  $D = 5$ , respectively. For all cases except the standard QAOA applied to  $D = 5$  graphs, we count the number of parameters and CNOTs needed to reach an energy error of  $\delta E = 10^{-3}$ . As standard QAOA for  $D = 5$  graphs never reaches this error threshold, we instead count the CNOT gates and parameters at the end of the simulation (15 layers). The dark (light) red bars show variational parameter (CNOT gate) counts. The error bars show variances obtained by sampling over 20 different instances of edge weights.

the solution much faster than its adiabatic counterpart, the standard QAOA. Before we present this evidence, we must first explain how  $\mathcal{H}_{CD}$  can be constructed using the concept of adiabatic gauge potentials.

### A. Approximate adiabatic gauge potentials

Here we briefly review the mathematical machinery of STA and adiabatic gauge potentials [38–40]. Let  $|\psi\rangle$  be a state evolving under  $\mathcal{H}(\theta(t))$ ,  $i\partial_t |\psi\rangle = \mathcal{H}(\theta(t)) |\psi\rangle$ , where  $\theta$  is a continuous variable that parameterizes the Hamiltonian. A unitary transformation  $U(\theta(t))$  can be applied to move the Hamiltonian  $\mathcal{H}(\theta(t))$  from the initial basis to its instantaneous eigenbasis, where  $\tilde{\mathcal{H}}(\theta) = U^\dagger(\theta)\mathcal{H}(\theta)U(\theta)$  is diagonal at all times. The Schrödinger equation in the instantaneous eigenbasis is  $i\partial_t |*\rangle \tilde{\psi} = [\tilde{\mathcal{H}} - \dot{\theta}\tilde{\mathcal{A}}_\theta] |*\rangle \tilde{\psi}$ , where  $|*\rangle \tilde{\psi} = U^\dagger |\psi\rangle$ ,  $\dot{\theta} = d\theta/dt$ , and  $\tilde{\mathcal{A}}_\theta = iU^\dagger \partial_\theta U$  is the adiabatic gauge potential

in the rotated frame. It is evident that the term  $-\dot{\theta}\tilde{\mathcal{A}}_\theta$  drives transitions between the energy levels of the original Hamiltonian  $\mathcal{H}$ . Therefore, one can add the counterdiabatic term  $\mathcal{H}_{CD} = \dot{\theta}\mathcal{A}_\theta$  to  $\mathcal{H}(\theta)$ , with  $\mathcal{A}_\theta = U\tilde{\mathcal{A}}_\theta U^\dagger$ , to eliminate such transitions in the rotated frame. This is the core of transitionless driving protocols.

Now, the matrix elements of the adiabatic gauge potential in the instantaneous eigenbasis are

$$\begin{aligned} \langle m(\theta) | \mathcal{A}_\theta | n(\theta) \rangle &= \langle m(\theta) | U \tilde{\mathcal{A}}_\theta U^\dagger | n(\theta) \rangle \\ &= i \langle m(\theta) | \partial_\theta U U^\dagger | n(\theta) \rangle \\ &= i \langle m(\theta) | \partial_\theta n(\theta) \rangle, \end{aligned} \quad (4)$$

where we used  $\tilde{\mathcal{A}}_\theta = iU^\dagger \partial_\theta U$  and  $|n(\theta)\rangle = U(\theta) |n_0\rangle$  with  $|n_0\rangle$  being independent of  $\theta$ . Moreover, the adiabatic gauge potential  $\mathcal{A}_\theta$  satisfies [22,38]

$$\langle m | \mathcal{A}_\theta | n \rangle = i \langle m | \partial_\theta n \rangle = i \frac{\langle m | \partial_\theta \mathcal{H} | n \rangle}{E_n - E_m}, \quad (5)$$

which is obtained by differentiating the eigenfunction  $\mathcal{H}(\theta) |n(\theta)\rangle = E_n(\theta) |n(\theta)\rangle$  with respect to  $\theta$ . Note that increasing the size of the system can lead to divergent matrix elements due to exponentially small denominators ( $E_n - E_m$ ). In this regard, Ref. [40] proposes an approximate gauge potential

$$\mathcal{A}_\theta^{(p)} = i \sum_{k=1}^p a_k [\mathcal{H}, \partial_\theta \mathcal{H}]_{2k-1}, \quad (6)$$

where  $[X, Y]_{k+1} = [X, [X, Y]_k]$  and  $\{a_1, a_2, \dots, a_p\}$  is a set of coefficients with  $p$  being the order of the expansion. This set of coefficients is found by minimizing  $\text{Tr}[G^2(\mathcal{A}_\theta^{(p)})]$ , where  $G(\mathcal{A}_\theta^{(p)}) = \partial_\theta \mathcal{H} - i[\mathcal{H}, \mathcal{A}_\theta^{(p)}]$  [40]. In fact,  $\text{Tr}[G^2(\mathcal{X})]$  is minimized when  $\mathcal{X}$  is equal to the exact adiabatic gauge potential  $\mathcal{A}_\theta$  [38,39]. Using matrix calculus identities and properties of the trace, it is straightforward to show that

$$\frac{\partial \text{Tr}[G^2(\mathcal{X})]}{\partial \mathcal{X}} = 2[\mathcal{H}, i\partial_\theta \mathcal{H} - [\mathcal{X}, \mathcal{H}]]. \quad (7)$$

Only adiabatic gauge potentials satisfy  $[\mathcal{H}, i\partial_\theta \mathcal{H} - [\mathcal{A}_\theta, \mathcal{H}]] = 0$ . This is easily proven by differentiating  $\tilde{\mathcal{H}}(\theta) = U^\dagger(\theta)\mathcal{H}(\theta)U(\theta)$  with respect to  $\theta$ ,

$$\partial_\theta \tilde{\mathcal{H}} = \partial_\theta U^\dagger U U^\dagger \mathcal{H} U + U^\dagger \partial_\theta \mathcal{H} U + U^\dagger \mathcal{H} U U^\dagger \partial_\theta U, \quad (8)$$

and noting that  $\tilde{\mathcal{A}}_\theta = -i\partial_\theta U^\dagger U = iU^\dagger \partial_\theta U$  and  $\mathcal{H} = \sum_n E_n(\theta) |n(\theta)\rangle \langle n(\theta)|$ , then

$$[\mathcal{A}_\theta, \mathcal{H}] = i(\partial_\theta \mathcal{H} - \sum_n \partial_\theta E_n(\theta) |n(\theta)\rangle \langle n(\theta)|). \quad (9)$$

Given that  $[\mathcal{H}, \sum_n \partial_\theta E_n(\theta) |n(\theta)\rangle \langle n(\theta)|] = 0$ , adiabatic gauge potentials clearly satisfy

$$[\mathcal{H}, i\partial_\theta \mathcal{H} - [\mathcal{A}_\theta, \mathcal{H}]] = 0. \quad (10)$$

### B. Connection between ADAPT-QAOA and STA

To investigate the connection between ADAPT-QAOA and STA, we apply the above formalism using the Hamiltonian  $\mathcal{H} = f(t)H_C + [1 - f(t)] \sum_i X_i$  with  $f(t) = t/T$  and, since there is no other continuous variable that parameterizes the

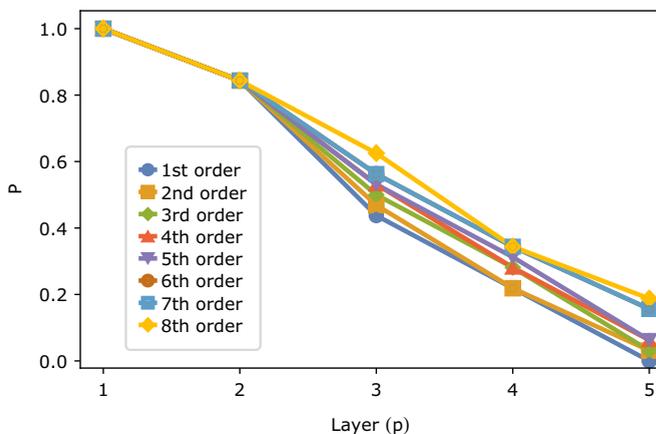


FIG. 3. Probability  $P$  of the operator at layer  $p$  of the ADAPT-QAOA Ansatz to be among the Pauli strings with the largest coefficient in  $\mathcal{H}_{CD}$  averaged over 32 graphs with  $n = 6$ ,  $D = 3$ . The different curves correspond to different orders of the approximation.

Hamiltonian, we simply set  $\theta = t$  in the equations above.  $T$  is the duration of the evolution from the initial state  $|\psi_{ref}\rangle = |+\rangle^{\otimes n}$  to the ground state of the cost Hamiltonian  $H_C$ , which is given by Eq. (3). The counterdiabatic Hamiltonian  $\mathcal{H}_{CD}$  is approximated using Eq. (6), where  $p$  is the order of the approximation.

As a concrete example, we study the Max-Cut problem on 32 instances of regular graphs ( $n = 6$ ,  $D = 3$ ) with random edge weights. Figure 3 shows the probability that an operator in the ADAPT-QAOA Ansatz is also one of the dominant operators in  $\mathcal{H}_{CD}$ . For each of the 32 cases, we define a set  $\mathcal{O}_{CD}^{(i)}$  (with  $i = 1, \dots, 32$ ) comprised of the five operators with the largest coefficient in the time-averaged  $\mathcal{H}_{CD}$ .<sup>1</sup> The probability  $P$  in Fig. 3 is constructed by taking the total number of times the mixer operator at layer  $p$  is also an element of the corresponding set  $\mathcal{O}_{CD}^{(i)}$  and dividing it by the total number of cases. In all cases, the mixer operator at the first layer is also an element of the set  $\mathcal{O}_{CD}^{(i)}$ . For higher layers, the probability of the mixer operator to be in  $\mathcal{O}_{CD}^{(i)}$  is inversely proportional to the layer number. We attribute this to the fact that  $\mathcal{H}_{CD}$  is computed for a specific mixer Hamiltonian ( $\sum_i^n X_i$ ), while information about this choice does not enter into ADAPT-QAOA, which only relies on the initial state  $|+\rangle^{\otimes n}$ .<sup>2</sup> Interestingly, from Fig. 3 we see that going to higher order in the  $\mathcal{H}_{CD}$  approximation increases the probability of finding the mixers in the set  $\mathcal{O}_{CD}^{(i)}$ . It therefore appears that ADAPT-QAOA finds the appropriate rotation axes in Hilbert space for faster convergence to the solution,

<sup>1</sup>We have seen in our simulations that only Pauli string operators are chosen by ADAPT-QAOA until deep into the layer number. Since we stop at layer five, only Pauli strings are chosen in all 32 cases.

<sup>2</sup>In principle, the mixer Hamiltonian in  $\mathcal{H}(t)$  could be replaced by any other Hamiltonian that has  $|+\rangle^{\otimes n}$  as the ground state, but this would modify the counterdiabatic Hamiltonian and any resemblance to the ADAPT-QAOA Ansatz would be reduced or even lost. Interestingly, using the mixer Hamiltonian  $\sum_i^n X_i$  involves the lowest possible energetic cost [41,42] of implementing the shortcut to adiabaticity.

and that these axes may in some sense be universal across all possible choices of  $\mathcal{H}(t)$  that interpolate between the initial and target states. This suggests that STA can be used as a tool to construct operator pools for ADAPT-QAOA.

#### IV. CONCLUSION

In conclusion, we introduced ADAPT-QAOA, an optimization algorithm that grows the Ansatz iteratively in a way that is naturally tailored to a given problem. We tested several instances of random diagonal Hamiltonians and found that ADAPT-QAOA always outperforms the standard QAOA. Given its flexibility with the choice of mixer pool, the algorithm can be tailored to the native gates, connectivities, and experimental constraints of hardware. It would also be fruitful to employ ADAPT-QAOA for optimization problems that use higher-dimensional Hilbert spaces, such as graph coloring [26,27]. Finally, more work into the connection to STA would be of both fundamental and practical interest.

#### ACKNOWLEDGMENTS

We thank Bryan T. Gard and Ada Warren for helpful discussions. S.E.E. acknowledges support from the US Department of Energy (Award No. DE-SC0019318). E.B. and N.J.M. acknowledge support from the US Department of Energy (Award No. DE-SC0019199).

#### APPENDIX A: ISING SYMMETRY AND MIXER POOL OPERATORS

In this work, we focus on Ising Hamiltonians of the form

$$H_C = -\frac{1}{2} \sum_{i,j} w_{i,j} (I - Z_i Z_j), \quad (\text{A1})$$

which have a  $Z_2$  symmetry associated with the operator  $F = \otimes_i X_i$ . Since  $[F, H_C] = 0$  and  $F|\psi_{ref}\rangle = |\psi_{ref}\rangle$ , we can rewrite the gradient in the first iteration as

$$\begin{aligned} & \langle \psi_{ref} | e^{iH_C \gamma} [H_C, A_j] e^{-iH_C \gamma} | \psi_{ref} \rangle \\ & = \langle \psi_{ref} | e^{iH_C \gamma} F [H_C, A_j] F e^{-iH_C \gamma} | \psi_{ref} \rangle, \end{aligned} \quad (\text{A2})$$

where  $A_j$  is an operator from the mixer pool. However, we also know that  $FA_j = \pm A_j F$  because  $F$  and  $A_j$  are Pauli strings (except when  $A_j$  is the standard QAOA mixer  $\sum_{i \in Q} X_i$  or  $\sum_{i \in Q} Y_i$ , but the former commutes and the latter anticommutes with  $F$ ), so  $F[H_C, A_j]F = \pm[H_C, A_j]$ . Comparing this to Eq. (A2), we see that to have a nonzero gradient, we need  $[F, A_j] \neq 0$ . This holds for all steps of the algorithm, because only operators that commute with  $F$  appear in the Ansatz, so a formula like Eq. (A2) holds at every iteration. The  $A_j$  that commute with  $F$  are Pauli strings that have an even number of  $Y$  or  $Z$  operators, so we retain only these Pauli strings in our mixer pool.

#### APPENDIX B: FIRST LAYER OF ADAPT-QAOA ANSATZ

Here we analyze the ADAPT-QAOA cost function in the first layer, and the results show that the minimum of the energy in the first layer of ADAPT-QAOA never occurs at  $\gamma = 0$  for

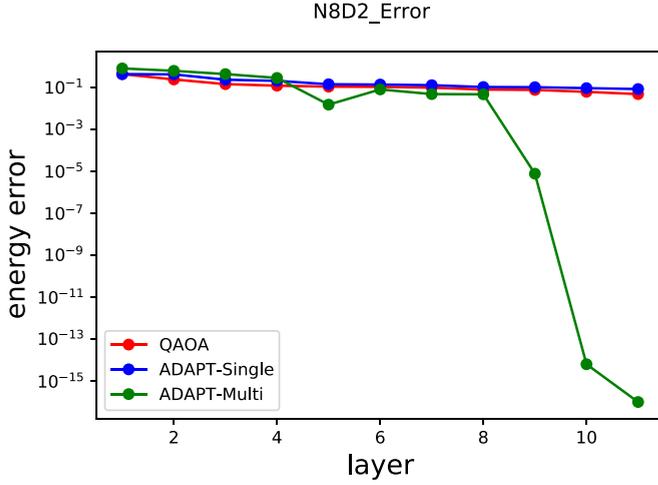


FIG. 4. The performance of three algorithms for the Max-Cut problem on a  $n = 8$  and  $D = 2$  graph with randomly chosen edge weights.

any operator included in the pool. We also show that  $\gamma = 0$  is a critical point of the cost function.

At level  $p$  of ADAPT-QAOA, the cost function is

$$E_p(\beta, \gamma) = \langle \psi^{(p-1)} | e^{-i\gamma H} e^{-i\beta M} H e^{i\beta M} e^{i\gamma H} | \psi^{(p-1)} \rangle. \quad (\text{B1})$$

Where  $H$  is a linear combination of Pauli strings that are tensor products of the identity and  $Z$ . All the terms in  $H$  commute with each other.  $M$  is the mixer. If the mixers  $M$  are single-Pauli strings, we have

$$e^{-i\beta M} H e^{i\beta M} = H_c + \cos(2\beta)H_a - i \sin(2\beta)MH_a, \quad (\text{B2})$$

where  $H_c$  is the part of  $H$  that commutes with  $M$ , and  $H_a$  is the part of  $H$  that anticommutes with  $M$ . We then have

$$\begin{aligned} e^{-i\gamma H} e^{-i\beta M} H e^{i\beta M} e^{i\gamma H} &= H_c + \cos(2\beta)H_a - i \sin(2\beta)e^{-i\gamma H} M e^{i\gamma H} H_a \\ &= H_c + \cos(2\beta)H_a - i \sin(2\beta)MH_a e^{2i\gamma H_a}. \end{aligned} \quad (\text{B3})$$

We know that  $E_p(\beta, \gamma)$  is periodic in  $\beta$  with period  $\pi$ . Therefore, we can restrict  $\beta$  to the range  $\beta \in [-\pi/2, \pi/2]$  without loss of generality. Let us define

$$G(\gamma) \equiv -iMH_a e^{2i\gamma H_a}. \quad (\text{B4})$$

The cost function is then

$$E_p(\beta, \gamma) = \langle H_c \rangle + \cos(2\beta)\langle H_a \rangle + \sin(2\beta)\langle G(\gamma) \rangle, \quad (\text{B5})$$

where the expectation values are taken with respect to  $|\psi^{(p-1)}\rangle$ . Therefore,

$$\begin{aligned} \frac{\partial E_p}{\partial \beta} &= -2 \sin(2\beta)\langle H_a \rangle + 2 \cos(2\beta)\langle G(\gamma) \rangle = 0 \\ \Rightarrow \tan(2\beta) &= \frac{\langle G(\gamma) \rangle}{\langle H_a \rangle}, \end{aligned} \quad (\text{B6})$$

and

$$\begin{aligned} \frac{\partial E_p}{\partial \gamma} &= \sin(2\beta)\langle G'(\gamma) \rangle = 0 \\ \Rightarrow \beta &= 0, \pm\pi/2 \quad \text{or} \quad \langle G'(\gamma) \rangle = 0. \end{aligned} \quad (\text{B7})$$

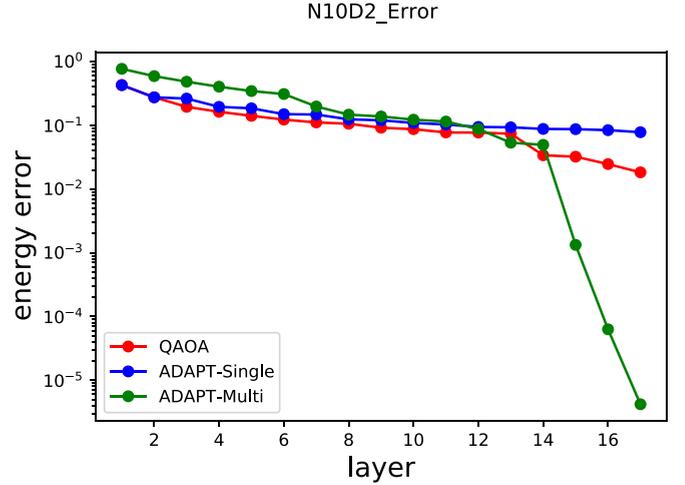


FIG. 5. The performance of three algorithms for the Max-Cut problem on a  $n = 10$  and  $D = 2$  graph with randomly chosen edge weights.

For the first layer,  $p = 1$ , the state is  $|\psi^{(0)}\rangle = |+\rangle^{\otimes n}$ , and so  $\langle H_a \rangle = 0$ . From Eq. (B6), we see that  $\beta = \pm\pi/4$  assuming  $\langle 0|G(\gamma)|0\rangle \neq 0$ . Eq. (B7) then requires  $\langle G'(\gamma) \rangle = 0$ . Using Eq. (B4), for  $p = 1$  this is

$$\begin{aligned} \langle 0|G'(\gamma)|0\rangle &= 2\langle 0|MH_a^2 e^{2i\gamma H_a}|0\rangle \\ &= 2\langle 0|H_a^2 e^{2i\gamma H_a}|0\rangle = 0. \end{aligned} \quad (\text{B8})$$

Notice that  $\gamma = 0$  is not a solution of this equation because  $\langle 0|H_a^2|0\rangle > 0$ , which follows from the fact that this is the norm of a nonzero state,  $H_a|+\rangle^{\otimes n}$ . Numerics are needed to determine if there is a nonzero value of  $\gamma$  that satisfies  $\langle 0|G'(\gamma)|0\rangle = 0$ .

On the other hand, if  $\langle 0|G(\gamma)|0\rangle = 0$ , then there is no constraint on  $\beta$  from  $\frac{\partial E_p}{\partial \beta} = 0$ . Notice that this happens when  $\gamma = 0$  since  $\langle 0|G(0)|0\rangle = -i\langle 0|MH_a|0\rangle = 0$ , which is true for regular graphs where  $H_a$  is a sum of terms like  $Z_j Z_k$ , and  $M$  is a Pauli string that commutes with  $F = \otimes_\ell X_\ell$ . In this case, we must have  $\beta = 0$  or  $\pm\pi/2$ . There could be other values of  $\gamma$  that satisfy  $\langle 0|G(\gamma)|0\rangle = 0$ , but it seems unlikely that both this condition and  $\langle 0|G'(\gamma)|0\rangle = 0$  can be satisfied simultaneously, so it is probably still true that  $\beta$  must be 0 or  $\pm\pi/2$  in this case.

In summary, there are two classes of possible solutions for  $p = 1$ : (i)  $\beta = \pm\pi/4$  and  $\gamma = \gamma^* \neq 0$  where  $\langle 0|G'(\gamma^*)|0\rangle = 0$ ; (ii)  $\gamma = \gamma^*$  where  $\langle 0|G(\gamma^*)|0\rangle = 0$  and either  $\langle 0|G'(\gamma^*)|0\rangle = 0$  or  $\beta = 0, \pm\pi/2$ . To clarify whether these solutions are maxima or minima, we calculated the second derivatives:

$$\frac{\partial^2 E_p}{\partial \beta^2} = -4 \cos(2\beta)\langle 0|H_a|0\rangle - 4 \sin(2\beta)\langle 0|G(\gamma)|0\rangle, \quad (\text{B9})$$

$$\frac{\partial^2 E_p}{\partial \gamma^2} = \sin(2\beta)\langle 0|G''(\gamma)|0\rangle, \quad (\text{B10})$$

$$\frac{\partial^2 E_p}{\partial \beta \partial \gamma} = 2 \cos(2\beta)\langle 0|G'(\gamma)|0\rangle. \quad (\text{B11})$$

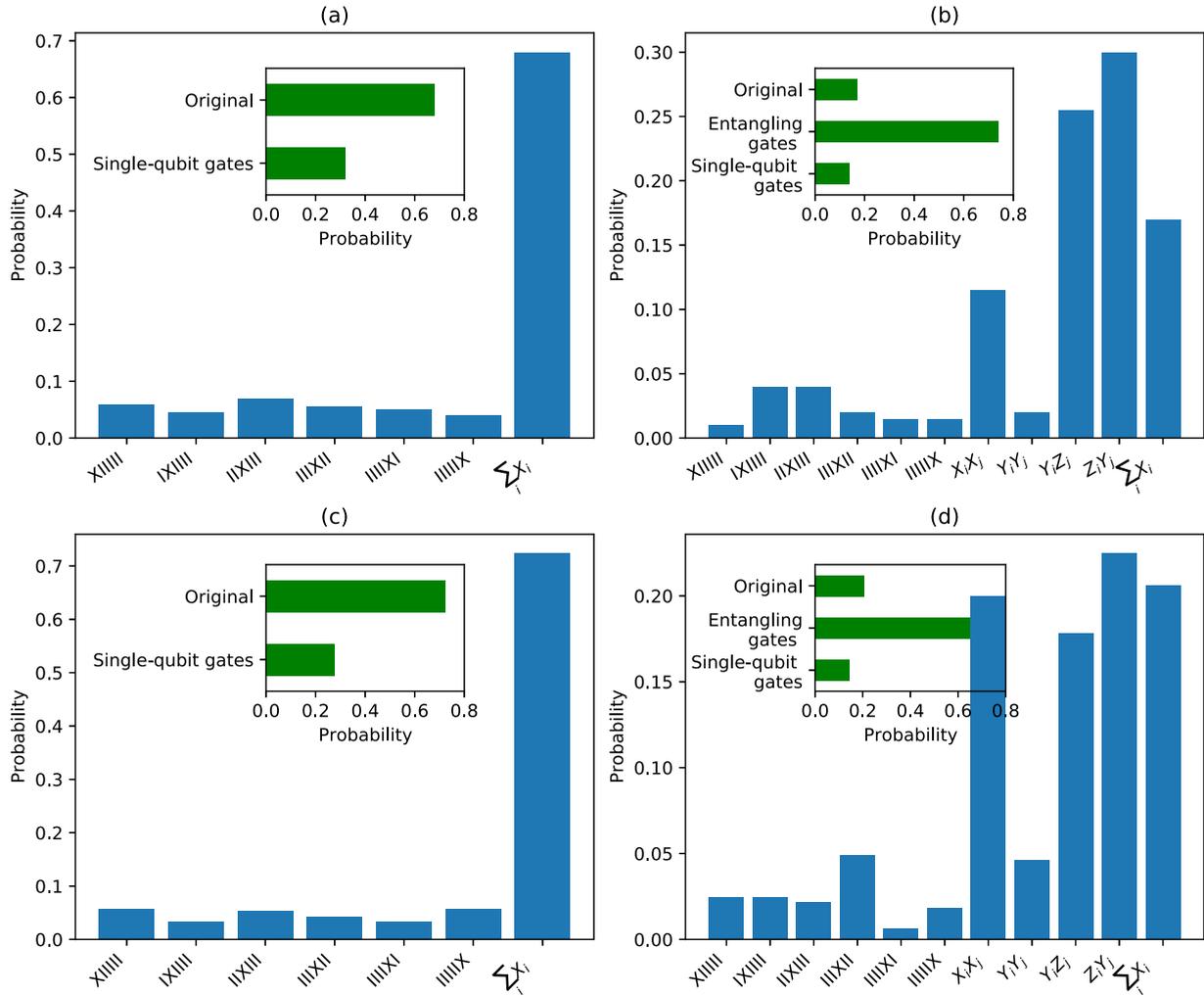


FIG. 6. Probability of operators picked by the original QAOA, ADAPT-QAOA with the single-qubit mixer and ADAPT-QAOA with multiqubit pool for the Max-Cut problem on regular graphs with  $n = 6$  vertices with degree (a), (b)  $D = 3$  and (c), (d)  $D = 5$  with random edge weights sampled from a uniform distribution  $U(0, 1)$ . The blue bars show the probability of each particular operator used for *Ansatz*, and green bars show the probability of the original mixer, sum over all single-qubit gates and sum over all entangling gates used in *Ansatz*. The results from 20 instances of random edge weights.

Let us consider the class (i) extrema first. In this case

$$\left. \frac{\partial^2 E_p}{\partial \beta^2} \right|_{\beta=\pm\pi/4, \gamma=\gamma^*} = \mp 4 \langle 0|G(\gamma^*)|0\rangle, \quad (B12)$$

and the Hessian is

$$\left\{ \frac{\partial^2 E_p}{\partial \beta^2} \frac{\partial^2 E_p}{\partial \gamma^2} - \left( \frac{\partial^2 E_p}{\partial \beta \partial \gamma} \right)^2 \right\} \Big|_{\beta=\pm\pi/4, \gamma=\gamma^*} = -4 \langle 0|G(\gamma^*)|0\rangle \langle 0|G''(\gamma^*)|0\rangle. \quad (B13)$$

Thus, we need  $\langle 0|G(\gamma^*)|0\rangle \langle 0|G''(\gamma^*)|0\rangle < 0$ , in which case  $\beta = \pi/4$  is a maximum and  $\beta = -\pi/4$  is a minimum if  $\langle 0|G(\gamma^*)|0\rangle > 0$ , while  $\beta = -\pi/4$  is a maximum and  $\beta = \pi/4$  is a minimum if  $\langle 0|G(\gamma^*)|0\rangle < 0$ .

For type (ii) extrema, we have

$$\left. \frac{\partial^2 E_p}{\partial \beta^2} \right|_{\beta=0, \pm\pi/2, \gamma=\gamma^*} = \pm 4 \langle 0|H_d|0\rangle = 0, \quad (B14)$$

and the Hessian is

$$\left\{ \frac{\partial^2 E_p}{\partial \beta^2} \frac{\partial^2 E_p}{\partial \gamma^2} - \left( \frac{\partial^2 E_p}{\partial \beta \partial \gamma} \right)^2 \right\} \Big|_{\beta=0, \pm\pi/2, \gamma=\gamma^*} = -4 \langle 0|G'(\gamma^*)|0\rangle^2, \quad (B15)$$

these extrema are saddle points. Notice that  $\gamma = 0$  is one of these saddle points.

We can see that the minimum of the energy in the first layer of ADAPT-QAOA never occurs at  $\gamma = 0$ , and the only minima occur at  $\beta = \pm\pi/4$  and  $\gamma = \gamma^* \neq 0$ , which is consistent with our numerical simulation results. The fact that  $\gamma = 0$  is a saddle point (at least in the first layer) makes this a problematic choice for the initial value of  $\gamma$  in the classical optimization subroutine of ADAPT-QAOA. For this reason, we instead choose the initial value,  $\gamma_0$ , to be a small nonzero value, e.g.,  $\gamma_0 = 0.01$ . Similar performance is achieved for any value of  $\gamma_0 \lesssim 0.1$ . Keeping  $\gamma_0$  close to zero is preferable to avoid biasing the optimization.

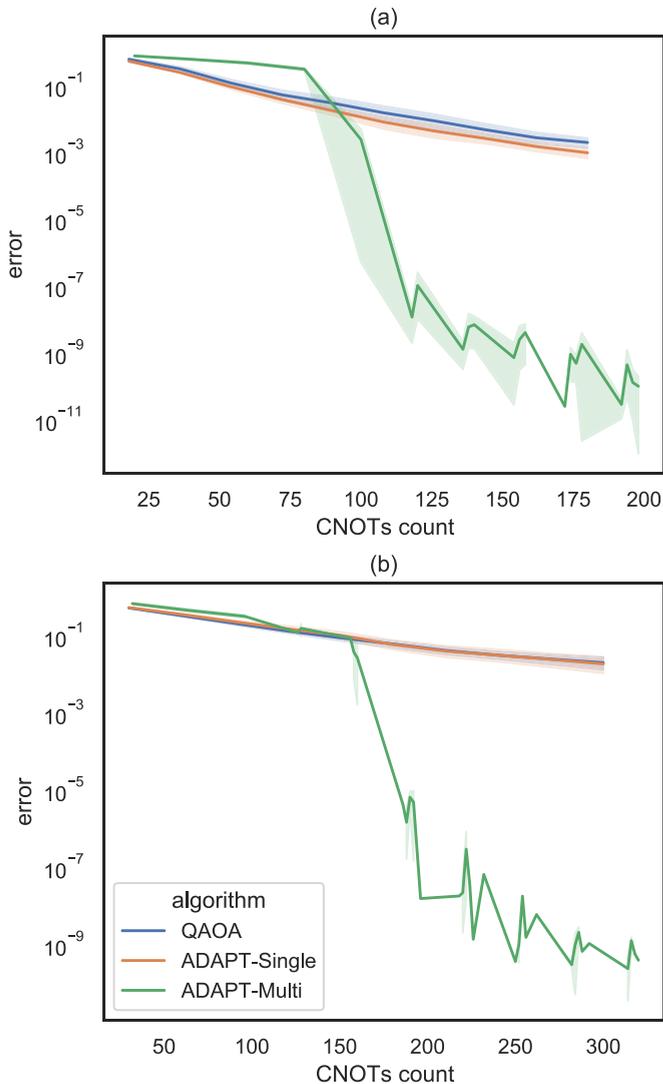


FIG. 7. Comparison of the performance of the original QAOA algorithm (blue) with the ADAPT-QAOA algorithm for the single-qubit (orange) and multiqubit (green) pools run for the Max-Cut problem on graphs with  $n = 6$  vertices with degree (a)  $D = 3$  and (b)  $D = 5$ . The energy error is shown as a function of the number of total CNOTs used in the *Ansatz*.

#### APPENDIX C: SCALABILITY OF ADAPT-QAOA

Here, we investigate the scalability of our algorithm. We perform simulations for degree  $D = 2$  graphs with  $n = 8$  and

$n = 10$  qubits, with results shown in Figs. 4 and 5. Clearly, even for larger system sizes and less connected graphs where  $D/n < 1/2$ , our algorithm exhibits substantially better performance compared to standard QAOA.

#### APPENDIX D: ROLE OF ENTANGLING MIXERS VERSUS ENTANGLING GATES

In Fig. 2, we compared the resources used by three different algorithms for the Max-Cut problem on regular graphs. These resources include the number of CNOT gates and the number of optimization parameters. From the comparison we can see that including entangling mixers in the *Ansatz* produces a dramatically faster convergence to the exact solution compared to the original QAOA. Surprisingly, despite the inclusion of these entangling mixers, the improvement in convergence comes with a simultaneous reduction in both the number of entangling gates in the compiled *Ansatz*.

To further investigate the role of entangling mixers, we consider the  $n = 6$ ,  $D = 3$  [in Figs. 6(a), 6(b)] and  $n = 6$ ,  $D = 5$  graphs [in Figs. 6(c), 6(d)] and show the probability for an operator to be picked by the original QAOA and by ADAPT-QAOA with a single-qubit mixer pool in Fig. S1. Similar results for ADAPT-QAOA with a multiqubit mixer pool are also shown in Figs. 6(b), 6(d). We find that when only the single-qubit mixer pool is used, the single-qubit operators  $X_i$  are chosen instead of the original mixer approximately 25% of the time. For the multiqubit mixer pool, the algorithm chooses two-qubit entangling operators approximately 70% of the time. Clearly, entangling mixers play a central role in the improved performance of ADAPT-QAOA.

Additionally, to understand the importance of CNOT gates in the compiled *Ansatz*, in Fig. 7 we show the error for the solution determined by each algorithm as a function of the number of CNOTs used in the *Ansatz* for both  $n = 6$ ,  $D = 3$ , and  $n = 6$ ,  $D = 5$  graphs. Based on the figure, we can see that ADAPT-QAOA with the multiqubit mixer pool converges using a much smaller number of CNOTs compared to the original QAOA or to ADAPT-QAOA with the single-qubit mixer pool. On the other hand, the latter only provides a modest reduction in CNOT count compared to the original QAOA. Interestingly, we see that while entangling mixers are crucial for the dramatic improvement in the performance afforded by ADAPT-QAOA with the multiqubit mixer pool, at the same time far fewer entangling gates are needed in the *Ansatz*. Thus, the role of entanglement in this algorithm is rather subtle.

- [1] Y.-H. Oh, H. Mohammadbagherpoor, P. Dreher, A. Singh, X. Yu, and A. J. Rindos, Solving multi-coloring combinatorial optimization problems using hybrid quantum algorithms, [arXiv:1911.00595](https://arxiv.org/abs/1911.00595).
- [2] G. Kochenberger, J.-K. Hao, F. Glover, M. Lewis, Z. Lü, H. Wang, and Y. Wang, The unconstrained binary quadratic programming problem: A survey, *J. Comb. Opt.* **28**, 58 (2014).
- [3] A. Lucas, Ising formulations of many NP problems, *Front. Phys.* **2**, 5 (2014).

- [4] E. Farhi, J. Goldstone, and S. Gutmann, A quantum approximate optimization algorithm, [arXiv:1411.4028](https://arxiv.org/abs/1411.4028).
- [5] E. Farhi, J. Goldstone, and P. Shor, Design of quantum algorithms using physics tools, Tech. Rep. 56291-PH-QC.10 (Massachusetts Institute of Technology, Cambridge, 2014).
- [6] E. Farhi and A. W. Harrow, Quantum supremacy through the quantum approximate optimization algorithm, [arXiv:1602.07674](https://arxiv.org/abs/1602.07674).
- [7] R. Shaydulin, I. Saffro, and J. Larson, Multistart methods for quantum approximate optimization, in *Proceedings of the IEEE*

- High Performance Extreme Computing Conference (HPEC), Waltham, MA, 2019* (IEEE, Piscataway, NJ, 2019).
- [8] G. Pagano, A. Bapat, P. Becker, K. Collins, A. De, P. Hess, H. Kaplan, A. Kyprianidis, W. Tan, C. Baldwin *et al.*, Quantum approximate optimization of the long-range ising model with a trapped-ion quantum simulator, *Proc. Natl. Acad. Sci.* **117**, 25396 (2020).
- [9] Z. Wang, S. Hadfield, Z. Jiang, and E. G. Rieffel, Quantum approximate optimization algorithm for MaxCut: A fermionic view, *Phys. Rev. A* **97**, 022304 (2018).
- [10] L. Zhou, S.-T. Wang, S. Choi, H. Pichler, and M. D. Lukin, Quantum Approximate Optimization Algorithm: Performance, Mechanism, and Implementation on Near-Term Devices, *Phys. Rev. X* **10**, 021067 (2020).
- [11] G. E. Crooks, Performance of the quantum approximate optimization algorithm on the maximum cut problem, [arXiv:1811.08419](https://arxiv.org/abs/1811.08419).
- [12] Z.-C. Yang, A. Rahmani, A. Shabani, H. Neven, and C. Chamon, Optimizing Variational Quantum Algorithms using Pontryagin's Minimum Principle, *Phys. Rev. X* **7**, 021027 (2017).
- [13] Z. Jiang, E. G. Rieffel, and Z. Wang, Near-optimal quantum circuit for Grover's unstructured search using a transverse field, *Phys. Rev. A* **95**, 062317 (2017).
- [14] P. K. Barkoutsos, G. Nannicini, A. Robert, I. Tavernelli, and S. Woerner, Improving variational quantum optimization using CVaR, *Quantum* **4**, 256 (2020).
- [15] G. G. Guerreschi and A. Matsuura, QAOA for Max-Cut requires hundreds of qubits for quantum speed-up, *Sci. Rep.* **9**, 6903 (2019).
- [16] M. Y. Niu, S. Lu, and I. L. Chuang, Optimizing QAOA: Success probability and runtime dependence on circuit depth, [arXiv:1905.12134](https://arxiv.org/abs/1905.12134).
- [17] M. E. S. Morales, J. D. Biamonte, and Z. Zimborás, On the universality of the quantum approximate optimization algorithm, *Quantum Info. Proc.* **19**, 291 (2020).
- [18] S. Lloyd, Quantum approximate optimization is computationally universal, [arXiv:1812.11075](https://arxiv.org/abs/1812.11075).
- [19] D. Wierichs, C. Gogolin, and M. Kastoryano, Avoiding local minima in variational quantum eigensolvers with the natural gradient optimizer, *Phys. Rev. Research* **2**, 043246 (2020).
- [20] S. Bravyi, A. Kliesch, R. Koenig, and E. Tang, Obstacles to Variational Quantum Optimization from Symmetry Protection, *Phys. Rev. Lett.* **125**, 260505 (2020).
- [21] M. Demirplak and S. A. Rice, Adiabatic population transfer with control fields, *J. Phys. Chem. A* **107**, 9937 (2003).
- [22] M. V. Berry, Transitionless quantum driving, *J. Phys. A: Math. Theor.* **42**, 365303 (2009).
- [23] D. Guéry-Odelin, A. Ruschhaupt, A. Kiely, E. Torrontegui, S. Martínez-Garaot, and J. G. Muga, Shortcuts to adiabaticity: Concepts, methods, and applications, *Rev. Mod. Phys.* **91**, 045001 (2019).
- [24] N. N. Hegade, K. Paul, Y. Ding, M. Sanz, F. Albarrán-Arriagada, E. Solano, and X. Chen, Shortcuts to Adiabaticity in Digitized Adiabatic Quantum Computing, *Phys. Rev. Appl.* **15**, 024038 (2021).
- [25] E. Farhi, J. Goldstone, S. Gutmann, and H. Neven, Quantum algorithms for fixed qubit architectures, [arXiv:1703.06199](https://arxiv.org/abs/1703.06199).
- [26] S. Hadfield, Z. Wang, B. O'Gorman, E. G. Rieffel, D. Venturelli, and R. Biswas, From the quantum approximate optimization algorithm to a quantum alternating operator Ansatz, *Algorithms* **12**, 34 (2019).
- [27] Z. Wang, N. C. Rubin, J. M. Dominy, and E. G. Rieffel, XY mixers: Analytical and numerical results for the quantum alternating operator Ansatz, *Phys. Rev. A* **101**, 012320 (2020).
- [28] H. R. Grimsley, S. E. Economou, E. Barnes, and N. J. Mayhall, An adaptive variational algorithm for exact molecular simulations on a quantum computer, *Nature Commun.* **10**, 3007 (2019).
- [29] H. L. Tang, V. O. Shkolnikov, G. S. Barron, H. R. Grimsley, N. J. Mayhall, E. Barnes, and S. E. Economou, qubit-adapt-vqe: An adaptive algorithm for constructing hardware-efficient ansätze on a quantum processor, *PRX Quantum* **2**, 020310 (2021).
- [30] W. J. Huggins, J. R. McClean, N. C. Rubin, Z. Jiang, N. Wiebe, K. B. Whaley, and R. Babbush, Efficient and noise resilient measurements for quantum chemistry on near-term quantum computers, *npj Quantum Info.* **7**, 23 (2021).
- [31] L. Li, M. Fan, M. Coram, P. Riley, and S. Leichenauer, Quantum optimization with a novel Gibbs objective function and Ansatz architecture search, *Phys. Rev. Research* **2**, 023074 (2020).
- [32] R. Shaydulin, I. Safro, and J. Larson, Multistart methods for quantum approximate optimization, in *2019 IEEE High Performance Extreme Computing Conference (HPEC)* (IEEE, New York, 2019), pp. 1–8.
- [33] M. P. Harrigan, K. J. Sung, M. Neeley, K. J. Satzinger, F. Arute, K. Arya, J. Atalaya, J. C. Bardin, R. Barends, S. Boixo *et al.*, Quantum approximate optimization of non-planar graph problems on a planar superconducting processor, *Nat. Phys.* **17**, 332 (2021).
- [34] V. Akshay, H. Philathong, M. E. S. Morales, and J. D. Biamonte, Reachability Deficits in Quantum Approximate Optimization, *Phys. Rev. Lett.* **124**, 090504 (2020).
- [35] C. Jarzynski, Generating shortcuts to adiabaticity in quantum and classical dynamics, *Phys. Rev. A* **88**, 040101(R) (2013).
- [36] J. Deng, Q.-h. Wang, Z. Liu, P. Hänggi, and J. Gong, Boosting work characteristics and overall heat-engine performance via shortcuts to adiabaticity: Quantum and classical systems, *Phys. Rev. E* **88**, 062122 (2013).
- [37] S. Iram, E. Dolson, J. Chiel, J. Pelesko, N. Krishnan, Ö. Güngör, B. Kuznets-Speck, S. Deffner, E. Ilker, J. G. Scott, and M. Hinczewski, Controlling the speed and trajectory of evolution with counterdiabatic driving, *Nature Phys.* **17**, 135 (2020).
- [38] M. Kolodrubetz, D. Sels, P. Mehta, and A. Polkovnikov, Geometry and non-adiabatic response in quantum and classical systems, *Phys. Rep.* **697**, 1 (2017).
- [39] D. Sels and A. Polkovnikov, Minimizing irreversible losses in quantum systems by local counterdiabatic driving, *Proc. Natl. Acad. Sci. USA* **114**, E3909 (2017).
- [40] P. W. Claeys, M. Pandey, D. Sels, and A. Polkovnikov, Floquet-Engineering Counterdiabatic Protocols in Quantum Many-Body Systems, *Phys. Rev. Lett.* **123**, 090602 (2019).
- [41] A. C. Santos and M. S. Sarandy, Superadiabatic controlled evolutions and universal quantum computation, *Sci. Rep.* **5**, 15775 (2015).
- [42] Y. Zheng, S. Campbell, G. De Chiara, and D. Poletti, Cost of counterdiabatic driving and work output, *Phys. Rev. A* **94**, 042132 (2016).