Quantum Approximate Optimization Algorithm: Performance, Mechanism, and Implementation on Near-Term Devices

Leo Zhou,[1,‡,*] Sheng-Tao Wang,[1,2,‡,†] Soonwon Choi,[1,3] Hannes Pichler,[4,1] and Mikhail D. Lukin[1]

[1] *Department of Physics, Harvard University, Cambridge, Massachusetts 02138, USA*
[2] *QuEra Computing Inc., Boston, Massachusetts 02135, USA*
[3] *Department of Physics, University of California Berkeley, Berkeley, California 94720, USA*
[4] *ITAMP, Harvard-Smithsonian Center for Astrophysics, Cambridge, Massachusetts 02138, USA*

(Received 9 November 2019; accepted 15 May 2020; published 24 June 2020)

The quantum approximate optimization algorithm (QAOA) is a hybrid quantum-classical variational algorithm designed to tackle combinatorial optimization problems. Despite its promise for near-term quantum applications, not much is currently understood about the QAOA's performance beyond its lowest-depth variant. An essential but missing ingredient for understanding and deploying the QAOA is a constructive approach to carry out the outer-loop classical optimization. We provide an in-depth study of the performance of the QAOA on MaxCut problems by developing an efficient parameter-optimization procedure and revealing its ability to exploit nonadiabatic operations. Building on observed patterns in optimal parameters, we propose heuristic strategies for initializing optimizations to find quasioptimal p-level QAOA parameters in $O[\mathrm{poly}(p)]$ time, whereas the standard strategy of random initialization requires $2^{O(p)}$ optimization runs to achieve similar performance. We then benchmark the QAOA and compare it with quantum annealing, especially on difficult instances where adiabatic quantum annealing fails due to small spectral gaps. The comparison reveals that the QAOA can learn via optimization to utilize nonadiabatic mechanisms to circumvent the challenges associated with vanishing spectral gaps. Finally, we provide a realistic resource analysis on the experimental implementation of the QAOA. When quantum fluctuations in measurements are accounted for, we illustrate that optimization is important only for problem sizes beyond numerical simulations but accessible on near-term devices. We propose a feasible implementation of large MaxCut problems with a few hundred vertices in a system of 2D neutral atoms, reaching the regime to challenge the best classical algorithms.

DOI: 10.1103/PhysRevX.10.021067

Subject Areas: Atomic and Molecular Physics,
Quantum Information,
Statistical Physics

I. INTRODUCTION

As quantum computing technology develops, there is a growing interest in finding useful applications of near-term quantum machines [1]. In the near future, however, the number of reliable quantum operations will be limited by noise and decoherence. As such, hybrid quantum-classical algorithms [2–4] have been proposed to make the best of available quantum resources and integrate them with classical routines. The quantum approximate optimization algorithm (QAOA) [2] and the variational quantum eigensolver [3] are such algorithms put forward to address classical combinatorial optimization and quantum chemistry problems, respectively. Proof-of-principle experiments running these algorithms have already been demonstrated in the lab [5–9].

In these hybrid algorithms, a quantum processor prepares a quantum state according to a set of variational parameters. Using measurement outputs, the parameters are then optimized by a classical computer and fed back to the quantum machine in a closed loop. In the QAOA, the state is prepared by a p-level circuit specified by $2p$ variational parameters. Even at the lowest circuit depth ($p = 1$), the QAOA has nontrivial provable performance guarantees [2,10] and is not efficiently simulatable by classical computers [11]. It is, thus, an appealing algorithm to explore quantum speedups on near-term quantum machines.

However, very little is known about the QAOA beyond the lowest depth. While the QAOA is known to monotonically improve with depth and succeed in the

[*] leozhou@g.harvard.edu
[†] swang@quera-computing.com
[‡] These authors contributed equally to this work.

$p \to \infty$ limit [2], its performance when $1 < p < \infty$ is largely unexplored. In fact, it has been argued that the QAOA needs to go beyond low depths in order to compete with the best classical algorithm for some problems on bounded-degree graphs [12,13]. It thus remains a critical problem to assess the QAOA at intermediate depths where one may hope for a quantum computational advantage. One major hurdle lies in the difficulty to efficiently optimize in the nonconvex, high-dimensional parameter landscape. Without constructive approaches to perform the parameter optimization, any potential advantages of the hybrid algorithms could be lost [14].

In this work, we contribute, in three major aspects, to the understanding and applicability of the QAOA on near-term devices, with a focus on MaxCut problems. First, we develop heuristic strategies to efficiently optimize QAOA variational parameters. These strategies are found, via extensive benchmarking, to be quasioptimal in the sense that they usually produce known global optima. The standard approach with random initialization generically requires $2^{O(p)}$ optimization runs to surpass our heuristics. Second, we benchmark the performance of the QAOA and compare it with quantum annealing. On difficult graph instances where the minimum spectral gap is very small, the time required for quantum annealing to remain adiabatic is very long, as it scales inversely with the square of the gap. For these instances, the QAOA is found to outperform adiabatic quantum annealing by multiple orders of magnitude in computation time. Last, we provide a detailed resource analysis on the experimental implementation of the QAOA with near-term quantum devices. Taking into account quantum fluctuations in projective measurements, we argue that optimization plays a role only for much larger problem sizes than numerically accessible ones. We also propose a 2D physical implementation of the QAOA on MaxCut with a few hundred Rydberg-interacting atoms, which can be put to the test against the best classical algorithm for potential quantum advantages.

Our main results can be summarized as follows. By performing extensive searches in the entire parameter space, we discover persistent patterns in the optimal parameters. Based on the observed patterns, we develop strategies for selecting initial parameters in optimization, which allow us to efficiently optimize the QAOA at a cost scaling polynomially in p. This scaling is in stark contrast to the $2^{O(p)}$ optimization runs required by random initialization approaches. We also propose a new parametrization of the QAOA that may significantly simplify optimization by reducing the dimension of the search space. Using our heuristic strategy, we benchmark the performance of the QAOA on many instances of MaxCut up to $N \leq 22$ vertices and level $p \leq 50$. Comparing the QAOA with quantum annealing, we find that the former can learn via optimization to utilize diabatic mechanisms [15–18] and overcome the challenges faced by adiabatic quantum annealing due to very small spectral gaps. Considering realistic experimental implementations, we also study the effects of quantum "projection noise" in measurement: We find that, for numerically accessible problem sizes, the QAOA can often obtain the solution among measurement outputs before the best variational parameters are found. Parameter optimization will be more useful at large system sizes (a few hundred vertices), as one expects the probability of finding the solution from projective measurements to decrease exponentially. At such system sizes, we analyze a procedure to make graphs more experimentally realizable by reducing the required range of qubit interactions via vertex renumbering. Finally, we discuss a specific implementation using neutral atoms interacting via Rydberg excitations [19,20], where a 2D implementation with a few hundred atoms appears feasible on a near-term device.

The rest of the paper is organized in the following order: In Sec. II, we review the QAOA and the MaxCut problem. In Sec. III, we describe some patterns found for QAOA optimal parameters and introduce heuristic optimization strategies based on the observed patterns. We benchmark our heuristic strategies and study the performance of the QAOA on typical MaxCut graph instances in Sec. IV. We then, in Sec. V, compare the QAOA with quantum annealing, shedding light on the nonadiabatic mechanism of the QAOA. Last, we discuss considerations for experimental implementations for large problem sizes in Sec. VI.

II. QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

Many interesting real-world problems can be framed as combinatorial optimization problems [21,22]. These are problems defined on N-bit binary strings $z = z_1 \ldots z_N$, where the goal is to determine a string that maximizes a given classical objective function $C(z)\colon \{+1,-1\}^N \mapsto \mathbb{R}_{\geq 0}$. An approximate optimization algorithm aims to find a string z that achieves a desired approximation ratio

$$\frac{C(z)}{C_{\max}} \geq r^*, \tag{1}$$

where $C_{\max} = \max_z C(z)$.

The QAOA is a quantum algorithm recently introduced to tackle these combinatorial optimization problems [2]. To encode the problem, the classical objective function can be converted to a quantum problem Hamiltonian by promoting each binary variable z_i to a quantum spin σ_i^z:

$$H_C = C(\sigma_1^z, \sigma_2^z, \ldots, \sigma_N^z). \tag{2}$$

For the p-level QAOA, which is visualized in Fig. 1(a), we initialize the quantum processor in the state $|+\rangle^{\otimes N}$ and then apply the problem Hamiltonian H_C and a mixing Hamiltonian $H_B = \sum_{j=1}^N \sigma_j^x$ alternately with controlled durations to generate a variational wave function

$$|\psi_p(\vec{\gamma},\vec{\beta})\rangle = e^{-i\beta_p H_B} e^{-i\gamma_p H_C} \ldots e^{-i\beta_1 H_B} e^{-i\gamma_1 H_C} |+\rangle^{\otimes N}, \quad (3)$$

which is parameterized by $2p$ variational parameters γ_i and β_i $(i = 1, 2, \ldots, p)$. We then determine the expectation value H_C in this variational state:

$$F_p(\vec{\gamma},\vec{\beta}) = \langle \psi_p(\vec{\gamma},\vec{\beta}) | H_C | \psi_p(\vec{\gamma},\vec{\beta}) \rangle, \quad (4)$$

which is done by repeated measurements of the quantum system in the computational basis. A classical computer is used to search for the optimal parameters $(\vec{\gamma}^*,\vec{\beta}^*)$ so as to maximize the averaged measurement output $F_p(\vec{\gamma},\vec{\beta})$:

$$(\vec{\gamma}^*,\vec{\beta}^*) = \arg \max_{\vec{\gamma},\vec{\beta}} F_p(\vec{\gamma},\vec{\beta}). \quad (5)$$

This search is typically done by starting with some initial guess of the parameters and performing simplex or gradient-based optimization. A figure of merit for benchmarking the performance of the QAOA is the approximation ratio

$$r = \frac{F_p(\vec{\gamma}^*,\vec{\beta}^*)}{C_{\max}}. \quad (6)$$

The framework of the QAOA can be applied to general combinatorial optimization problems. Here, we focus on its application to an archetypical problem called MaxCut, which is a combinatorial problem whose approximate optimization beyond a minimum ratio r^* is NP-hard [23,24]. The MaxCut problem, visualized in Fig. 1(b), is defined for any input graph $G = (V, E)$. Here, $V = \{1, 2, \ldots, N\}$ denotes the set of vertices, and $E = \{(\langle i, j \rangle, w_{ij})\}$ is the set of edges, where $w_{ij} \in \mathbb{R}_{\geq 0}$ is the weight of the edge $\langle i, j \rangle$ connecting vertices i and j. The goal of MaxCut is to maximize the following objective function:

$$H_C = \sum_{\langle i,j \rangle} \frac{w_{ij}}{2} (\mathbb{1} - \sigma_i^z \sigma_j^z), \quad (7)$$

where an edge $\langle i, j \rangle$ contributes with weight w_{ij} if and only if spins σ_i^z and σ_j^z are antialigned.

For simplicity, we restrict our attention to MaxCut on d-regular graphs, where every vertex is connected to exactly d other vertices. We study two classes of graphs: The first is unweighted d-regular graphs (udR), where all edges have equal weights $w_{ij} = 1$; the second is weighted d-regular graphs (wdR), where the weights w_{ij} are chosen uniformly at random from the interval [0, 1]. It is NP-hard to design an algorithm that guarantees a minimum approximation ratio of $r^* \geq 16/17$ for MaxCut on all graphs [23] or $r^* \geq 331/332$ when restricted to u3R graphs [24]. The current record for approximation ratio guarantee on generic graphs belongs to Goemans and Williamson [25], achieving $r^* \approx 0.87856$ with semidefinite programing. This lower bound can be raised to $r^* \approx 0.9326$ when restricted to u3R graphs [26]. Farhi, Goldstone, and Gutmann [2] were able to prove that the QAOA at level $p = 1$ achieves $r^* \geq 0.6924$ for u3R graphs, using the fact that F_p can be written as a sum of quasilocal terms, each corresponding to a subgraph involving edges at most p steps away from a given edge. However, this approach to bound r^* quickly becomes intractable, since the locality of each term (i.e., size of each subgraph) grows exponentially in p, as does the number of subgraph types involved.

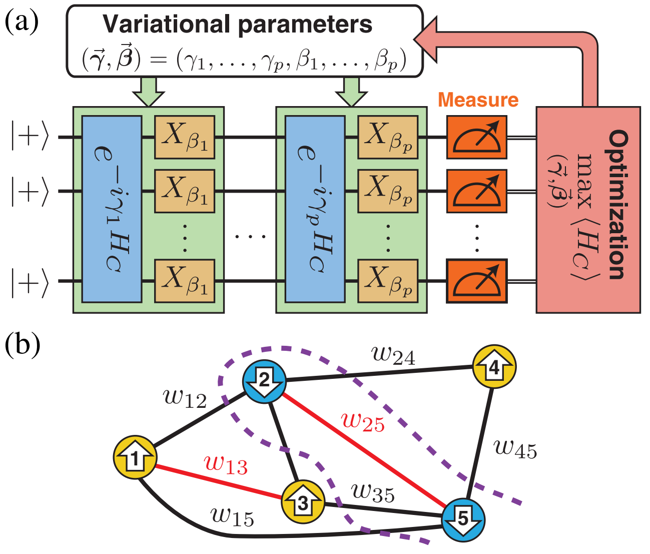

FIG. 1. (a) Schematic of a p-level quantum approximate optimization algorithm [2]. A quantum circuit takes input $|+\rangle^{\otimes n}$ and alternately applies $e^{-i\gamma_i H_C}$ and $X_{\beta_i} = e^{-i\beta_i \sigma^x}$, and the final state is measured to obtain the expectation value with respect to the objective function H_C. This result is fed to a (classical) optimizer to find the best parameters $(\vec{\gamma},\vec{\beta})$ that maximizes $\langle H_C \rangle$. (b) An example of a MaxCut problem on a five-vertex graph, where one seeks an assignment of spin variables on the vertices such that the sum of edge weights between antialigned spins is maximized (black edges).

The QAOA is believed to be a promising algorithm for multiple reasons [2,10,11,27–32]. As mentioned above, for certain cases, one can prove a guaranteed minimum approximation ratio when $p = 1$ [2,10]. Additionally, under reasonable complexity-theoretic assumptions, the QAOA cannot be efficiently simulated by any classical computer even when $p = 1$, making it a suitable candidate algorithm to establish the so-called "quantum supremacy" [11]. It has also been argued that the square-pulse ("bang-bang") ansatz of dynamical evolution, of which the QAOA is an example, is optimal given a fixed quantum computation time [29]. In general, the performance of the QAOA can only improve with increasing p, achieving $r \to 1$ when

$p \to \infty$, since it can approximate adiabatic quantum annealing via Trotterization; this monotonicity makes it more attractive than quantum annealing, whose performance may decrease with an increased run time [15].

While the QAOA has a simple description, not much is currently understood beyond $p = 1$. To establish a potential quantum advantage over classical algorithms, it is of critical importance to investigate the QAOA at intermediate depths ($p > 1$). References [2,12,13] have shown that the QAOA has limited performance on some problems on bounded-degree graphs when the depth is shallow. This limitation may result from the fact that the algorithm cannot "see" the entire graph at a low depth. It thus indicates that one may need the depth of the QAOA to grow with the system size (e.g., $p \geq \log N$) in order to outperform the best classical algorithms. For the toy example of MaxCut on u2R graphs, i.e., 1D antiferromagnetic rings, it is conjectured that the QAOA yields $r \geq (2p+1)/(2p+2)$ based on numerical evidence [2,27,33]. In another example of Grover's unstructured search problem among n items, the QAOA is shown to be able to find the target state with $p = \Theta(\sqrt{n})$, achieving the optimal query complexity within a constant factor [31]. For more general problems, Farhi, Goldstone, and Gutmann [2] propose a simple approach by discretizing each parameter into $O[\mathrm{poly}(N)]$ grid points; this approach, however, requires examining $N^{O(p)}$ possibilities at level p, which quickly becomes impractical as p grows. Efficient optimization of QAOA parameters and understanding the algorithm for $1 < p < \infty$ remain outstanding problems. We address these problems in the present work.

III. OPTIMIZING VARIATIONAL PARAMETERS

In this section, we address the issue of parameter optimization in QAOA, since searching for the best parameters via standard approaches that rely on random initialization generally become exponentially difficult as level p increases. We mostly restrict our discussion to randomly generated instances of u3R and w3R graphs. Similar results are found for u4R and w4R graphs, as well as complete graphs with random weights. We utilize patterns in the optimal parameters to develop heuristic strategies that can efficiently find quasi-optimal solutions in $O(\mathrm{poly}(p))$ time.

A. Patterns in optimal parameters

Before searching for patterns in optimal QAOA parameters, it is useful to eliminate degeneracies in the parameter space due to symmetries. Generally, QAOA has a time-reversal symmetry, $F_p(\vec{\gamma},\vec{\beta}) = F_p(-\vec{\gamma},-\vec{\beta})$, since both H_B and H_C are real-valued. For QAOA applied to MaxCut, there is an additional $\mathbb{Z}_2$ symmetry, as $e^{-i(\pi/2)H_B} \equiv (\sigma^x)^{\otimes N}$ commutes through the circuit. Furthermore, the structure of the MaxCut problem on udR graphs creates redundancy since $e^{-i\pi H_C} = \mathbb{1}$ if d is even, and $(\sigma^z)^{\otimes N}$ if d is odd. These symmetries allow us to restrict $\beta_i \in [-(\pi/4),(\pi/4))$ in general, and $\gamma_i \in [-(\pi/2),(\pi/2))$ for udR graphs.

We start by numerically investigating the optimal QAOA parameters for MaxCut on random u3R and w3R graphs with vertex number $8 \leq N \leq 22$, with extensive searches in the entire parameter space. For each graph instance and a given level p, we choose a random initial point (seed) in the parameter space [34] and use a commonly used, gradient-based optimization routine known as Broyden-Fletcher-Goldfarb-Shanno (BFGS) [35] to find a local optimum $(\vec{\gamma}^L,\vec{\beta}^L)$. This local optimization is repeated with sufficiently many different seeds to find the global optimum $(\vec{\gamma}^*,\vec{\beta}^*)$ [36]. We then reduce the degeneracies of the optimal parameters $(\vec{\gamma}^*,\vec{\beta}^*)$ using the aforementioned symmetries. In all cases examined, we find that the global optimum is nondegenerate up to these symmetries.

After performing the above numerical experiment for 100 random u3R and w3R graphs with various vertex numbers N, we discover some patterns in the optimal parameters $(\vec{\gamma}^*,\vec{\beta}^*)$. Generically, the optimal γ_i^* tends to increase smoothly with $i = 1,2,\ldots,p$, while β_i^* tends to decrease smoothly, as shown for the example instance in Fig. 2(a). In Fig. 2(b), we illustrate the pattern by simultaneously plotting the optimal parameters for 40 instances of

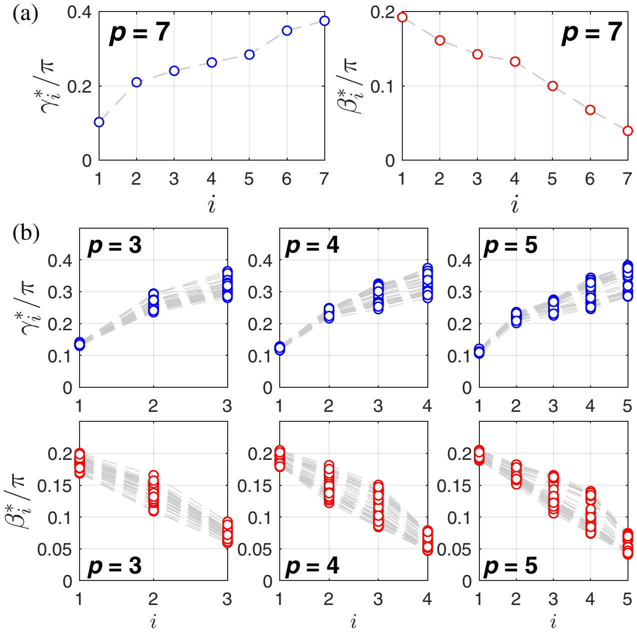

FIG. 2. (a) Optimal QAOA parameters $(\vec{\gamma}^*,\vec{\beta}^*)$ for an example instance of MaxCut on a 16-vertex unweighted 3-regular (u3R) graph at level $p = 7$. (b) The parameter pattern visualized by plotting the optimal parameters of 40 instances of 16-vertex u3R graphs, for $3 \leq p \leq 5$. Each dashed line connects parameters for one particular graph instance. For each instance and each p, we use the classical BFGS optimization routine [35] from 10^4 random initial points, and keep the best parameters.

16-vertex u3R graphs for $3 \leq p \leq 5$. Furthermore, for a given class of graphs, the optimal parameters are observed to roughly occupy the same range of values as p is varied. Similar patterns are found for w3R graphs and weighted complete graphs, which we illustrate in the Appendix A. These observations show a clear pattern in the optimal QAOA parameters that we can exploit in the optimization, as we discuss later in Sec. III B. Similar patterns are found for parameters up to $p \lesssim 15$, if the number of random seeds is increased accordingly.

We give two remarks here: First, we note that surprisingly, even at a small depth, this parameter pattern is reminiscent of adiabatic quantum annealing where H_C is gradually turned on while H_B is gradually turned off. However, we demonstrate in Sec. V that the mechanism of the QAOA goes beyond the adiabatic principle. Second, we note that these optimal parameters have a small spread over many different instances. This low variance is because the objective function $F_p(\vec{\gamma}, \vec{\beta})$ is a sum of terms corresponding to subgraphs involving vertices that are a distance $\leq p$ away from every edge. At small p, there are only a few relevant subgraph types that enter into F_p and effectively determine the optimal parameters. As $N \to \infty$ and at a fixed finite p, we expect the probability of a relevant subgraph type appearing in a random graph to approach a fixed fraction. This asymptotic behavior implies that the distribution of optimal parameters $(\vec{\gamma}^*, \vec{\beta}^*)$ should converge to a fixed set of values in this limit.

B. Heuristic optimization strategy for large *p*

The optimal parameter patterns observed above indicate that, generically, there is a slowly varying continuous curve that underlies the parameters γ_i^* and β_i^*. Moreover, this curve changes only slightly from level p to $p+1$. These observations allow us to choose educated guesses of variational parameters for the $(p+1)$-level QAOA based on optimized parameters at level p (or, in general, at level q, where $q \leq p$). These educated guesses can serve as initial points fed to various classical optimization routines that find a nearby local optimum. Based on this idea, we develop two types of heuristic strategies for initializing optimization. We give a high-level overview of these heuristics in this section while deferring the details of its implementation to Appendix B. While these heuristics are not guaranteed to find the global optimum of QAOA parameters, we show in Sec. IV A that it can produce, in $O[\mathrm{poly}(p)]$ time, quasioptima that require $2^{O(p)}$ randomly initialized optimization runs to surpass. Consequently, these heuristics allow us to study the performance and mechanism of the QAOA beyond $p = 1$.

The first heuristic strategy, which we call INTERP, uses linear interpolation to choose initial parameters. Starting at level $p = 1$, we optimize the parameters at level p, and then linearly interpolate the curve formed by them to extract a set of initial parameters for level $p+1$.

The second heuristic strategy, which we call FOURIER, uses a new parameterization of the QAOA. Instead of using the $2p$ parameters $(\vec{\gamma}, \vec{\beta}) \in \mathbb{R}^{2p}$, we switch to $2q$ parameters $(\vec{u}, \vec{v}) \in \mathbb{R}^{2q}$, where the individual elements γ_i and β_i are written as functions of $(\vec{u}, \vec{v})$ through the following transformation:

$$\gamma_i = \sum_{k=1}^{q} u_k \sin\left[\left(k - \frac{1}{2}\right)\left(i - \frac{1}{2}\right)\frac{\pi}{p}\right],$$
$$\beta_i = \sum_{k=1}^{q} v_k \cos\left[\left(k - \frac{1}{2}\right)\left(i - \frac{1}{2}\right)\frac{\pi}{p}\right]. \tag{8}$$

These transformations are known as discrete sine and cosine transforms, where u_k and v_k can be interpreted as the amplitude of the kth frequency component for $\vec{\gamma}$ and $\vec{\beta}$, respectively. In this strategy, when optimizing level $p+1$, the initial parameters are generated by simply reusing the optimized amplitudes $(\vec{u}^*, \vec{v}^*)$ from level p. Note that, when $q \geq p$, the $(\vec{u}, \vec{v})$ parametrization is capable of describing all possible QAOA protocols at level p. However, the smoothness of the optimal parameters $(\vec{\gamma}, \vec{\beta})$ implies that only the low-frequency components are important. Thus, we can also consider the case where q is a fixed constant independent of p, so the number of parameters is bounded even as the QAOA circuit depth increases.

While both heuristics work well, we decide to focus on using the FOURIER heuristics in the main text, because we find that it gives a slight edge in performance for MaxCut problems. The INTERP heuristic is simpler and may work better on other problems. More details on the implementation of the heuristics and their comparison can be found in Appendix B.

We stress that these heuristic strategies are developed to generate good initial points for optimization. These initial points can then be fed to standard optimization routines such as gradient descent, BFGS [35], Nelder-Mead [37], and Bayesian optimization [38]. This approach is in contrast to the standard strategy of random initialization (RI), where one picks a random set of parameters to begin optimization. In order to find the global optimum in a highly nonconvex landscape, the number of RI runs needed generically scales exponentially with the number of parameters, which becomes intractable for a large number of parameters. In the following section, we compare our heuristics to the RI approach and find that an exponential number of RI runs is needed to match the performance of our heuristics.

IV. PERFORMANCE OF HEURISTICALLY OPTIMIZED QAOA

A. Comparison between our heuristics and RI optimizations

Here, we compare the performance of our heuristic strategies to the standard strategy of random initialization.

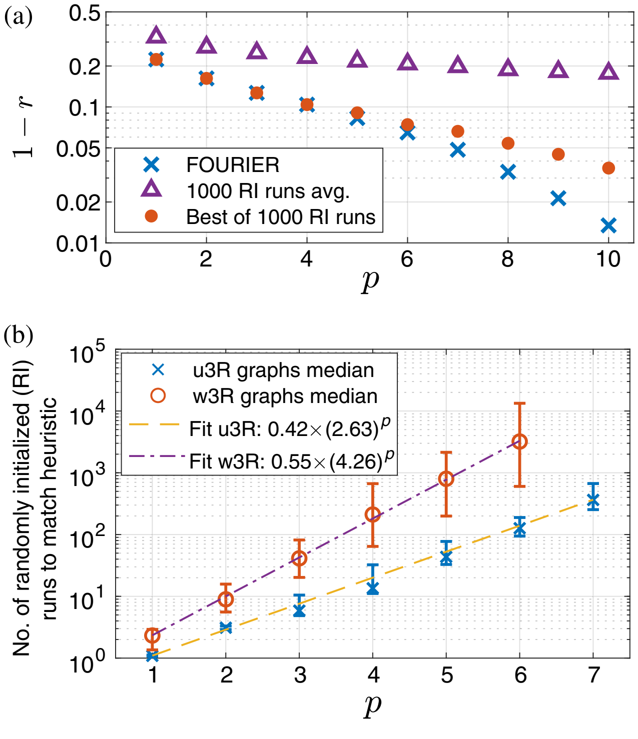

FIG. 3. (a) Comparison between our FOURIER heuristic and the RI approach for optimizing the QAOA, on an example instance of a 16-vertex w4R graph. The figure of merit $1-r$, where r is the approximation ratio, is plotted as a function of QAOA level p on a log-linear scale. The RI points are obtained by optimizing from 1000 randomly generated initial parameters, averaged over ten such realizations. (b) The median number of randomly initialized optimization runs needed to obtain an approximation ratio as good as our FOURIER heuristic, for 40 instances of 16-vertex u3R and w3R graphs. A log-linear scale is used, and exponential curves are fitted to the results. Error bars are 5th and 95th percentiles.

In Fig. 3(a), we show the result for an example instance of a 16-vertex w4R graph. At low p, our FOURIER heuristic strategies perform just as well as the best out of 1000 RI optimization runs—both are able to find the global optimum. The average performance of the RI strategy, on the other hand, is much worse than our heuristics. This result indicates that the QAOA parameter landscape is highly nonconvex and filled with low-quality, nondegenerate local optima. When $p \geq 5$, our heuristic optimization outperforms the best RI run. To estimate the number of RI runs needed to find an optimum with the same or better approximation ratio as our FOURIER heuristics, we generate 40 instances of 16-vertex u3R and w3R graphs and perform 40 000 RI optimization for each instance at each level p. In Fig. 3(b), the median number of RI runs needed to match our heuristic is shown to scale exponentially with p. Therefore, our heuristics offer a dramatic improvement in the resource required to find good parameters for the QAOA. As we verify with an excessive number of RI runs, the heuristics usually find the global optima.

We remark that, although we mostly use a gradient-based optimization routine (BFGS) in our numerical simulations, nongradient-based routines, such as Nelder-Mead [37], work equally well with our heuristic strategies. The choice to use BFGS is mainly motivated by the simulation speed. Later, in Sec. VI, we account for the measurement costs in estimating the gradient using a finite-difference method and perform a full Monte Carlo simulation of the entire QAOA algorithm, including quantum fluctuations in the determination of F_p.

B. Performance of QAOA on typical instances

With our heuristic optimization strategies in hand, we study the performance of the intermediate p-level QAOA on many graph instances. We consider many randomly generated instances u3R and w3R graphs with vertex number $8 \leq N \leq 22$ and use our FOURIER strategy to find optimal QAOA parameters for level $p \leq 20$. In the following discussion, we use the fractional error $1-r$ to assess the performance of the QAOA.

We first examine the case of unweighted graphs, specifically u3R graphs. In Fig. 4(a), we plot the fractional error $1-r$ as a function of the QAOA's level p. Here, we see that, on average, $1-r \propto e^{-p/p_0}$ appears to decay exponentially with p. Note that, since the instances studied are u3R graphs with system size $N \leq 22$, where $C_{\max} \leq |E| \leq 33$, we would have essentially prepared the MaxCut state whenever the fractional error goes below 1/33. This good performance can be understood by interpreting the QAOA as Trotterized quantum annealing, especially when the optimized parameters are of the pattern seen in Fig. 2, where one initializes in the ground state of $-H_B$ and evolves with H_B (and H_C) with smoothly decreasing (and increasing) durations. The equivalent total annealing time T is approximately proportional to the level p, since the individual parameters $\gamma_i, \beta_i = O(1)$ and correspond to the evolution times under H_C and H_B. If T is much longer than $1/\Delta_{\min}^2$, where $\Delta_{\min}$ is the minimum spectral gap, quantum annealing is able to find the exact solution to MaxCut (ground state of $-H_C$) by the adiabatic theorem [18] and achieves an exponentially small fractional error as predicted by Landau and Zener [39]. Numerically, we find that the minimum gaps of these u3R instances when running quantum annealing are on the order of $\Delta_{\min} \gtrsim 0.2$. It is thus not surprising that the QAOA achieves an exponentially small fractional error on average, since it is able to prepare the ground state of $-H_C$ through the adiabatic mechanism for these large-gap instances. Nevertheless, as we see in the following section, this exponential behavior breaks down for hard instances, where the gap is small.

Let us now turn to the case of w3R graphs. As shown in Fig. 4(b), the fractional error appears to scale as $1-r \propto e^{-\sqrt{p/p_0}}$. We note that the stretched-exponential scaling is true in the average sense, while individual instances have

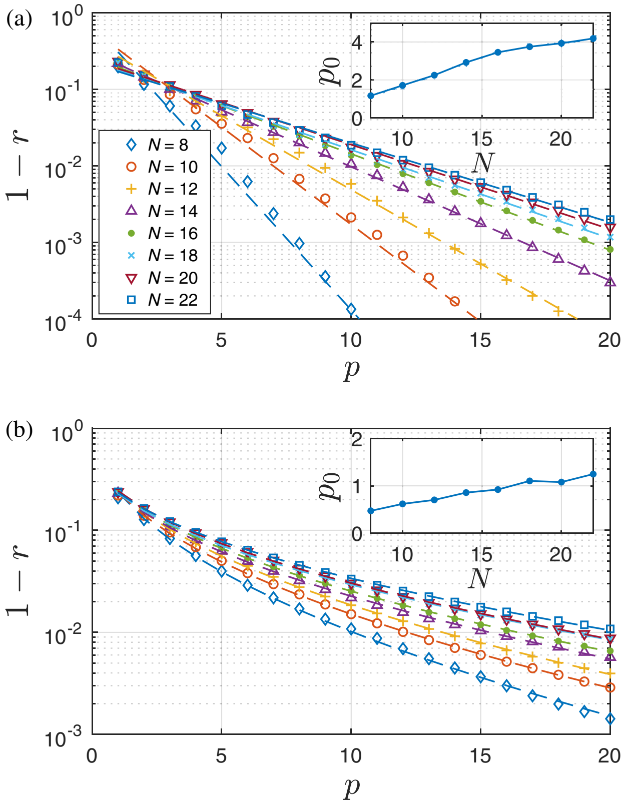

FIG. 4. Average performance of the QAOA as measured by the fractional error $1 - r$, plotted on a log-linear scale. The results are obtained by applying our heuristic optimization strategies FOURIER to up to 100 random instances of (a) u3R graphs and (b) w3R graphs. Differently colored lines correspond to fitted lines to the average for different system size N, where the model function is $1 - r \propto e^{-p/p_0}$ for unweighted graphs and $1 - r \propto e^{-\sqrt{p/p_0}}$ for weighted graphs. The insets show the dependence of the fit parameters p_0 on the system size N.

very different behavior. For easy instances whose corresponding minimum gaps $\Delta_{\min}$ are large, exponential scaling of the fractional error is found. For more difficult instances whose minimum gaps are small, we find that their fractional errors reach plateaus at intermediate p, before decreasing further when p is increased. We analyze these hard instances in more depth in the following Sec. V. Surprisingly, when we average over randomly generated instances, the fractional error is fitted remarkably well by a stretched-exponential function.

These results of average performance of the QAOA are interesting despite considerations of finite-size effects. While the decay constant p_0 does appear to depend on the system size N as shown in the insets in Fig. 4, our finite-size simulations cannot conclusively determine the exact scaling. Although it remains unknown whether the (stretched-)exponential scaling will start to break down at larger system sizes, if the trend continues to a system size of $N = 100$–1000, then the QAOA will be practically useful in solving interesting MaxCut instances, better or on par with other known algorithms, in a regime where finding the exact solution will be infeasible. Even if the QAOA fails for the worst-case graphs, it can still be practically useful, if it performs well on a randomly chosen graph of large size.

V. ADIABATIC MECHANISM, QUANTUM ANNEALING, AND QAOA

In the previous section, we benchmark the performance of the QAOA for MaxCut on random graph instances in terms of the approximation ratio r. Although designed for approximate optimization, the QAOA is often able to find the MaxCut configuration—the global optimum of the problem—with a high probability as level p increases. In this section, we assess the efficiency of the algorithm to find the MaxCut configuration and compare it with quantum annealing. In particular, we find that the QAOA is not necessarily limited by the minimum gap as in quantum annealing and explain a mechanism at work that allows it to overcome the adiabatic limitations.

A. Comparing QAOA with quantum annealing

A predecessor of the QAOA, quantum annealing (QA) is widely studied for the purpose of solving combinatorial optimization problems [40–43]. To find the MaxCut configuration that maximizes $\langle H_C \rangle$, we consider the following simple QA protocol:

$$H_{\mathrm{QA}}(s) = -[sH_C + (1 - s)H_B], \qquad s = t/T, \tag{9}$$

where $t \in [0, T]$ and T is the total annealing time. The initial state is prepared to be the ground state of $H_{\mathrm{QA}}(s = 0)$, i.e., $|\psi(0)\rangle = |+\rangle^{\otimes N}$. The ground state of the final Hamiltonian, $H_{\mathrm{QA}}(s = 1)$, corresponds to the solution of the MaxCut problem encoded in H_C [44]. In adiabatic QA, the algorithm relies on the adiabatic theorem to remain in the instantaneous ground state along the annealing path and solves the computational problem by finding the ground state at the end. To guarantee success, the necessary run time of the algorithm typically scales as $T = O(1/\Delta_{\min}^2)$, where $\Delta_{\min}$ is the minimum spectral gap [18]. Consequently, adiabatic QA becomes inefficient for instances where $\Delta_{\min}$ is extremely small. We refer to these graph instances as hard instances (for adiabatic QA).

Beyond the completely adiabatic regime, there is often a trade-off between the success probability [ground state population $p_{\mathrm{GS}}(T)$] and the run time (annealing time T): One can either run the algorithm with a long annealing time to obtain a high success probability or run it multiple times at a shorter time to find the solution at least once. A metric often used to determine the best balance is the time to solution (TTS) [18]:

$$\mathrm{TTS}_{\mathrm{QA}}(T) = T\frac{\ln(1-p_d)}{\ln[1-p_{\mathrm{GS}}(T)]}, \tag{10}$$

$$\mathrm{TTS}_{\mathrm{QA}}^{\mathrm{opt}} = \min_{T>0} \mathrm{TTS}_{\mathrm{QA}}(T). \tag{11}$$

$\mathrm{TTS}_{\mathrm{QA}}(T)$ measures the time required to find the ground state at least once with the target probability p_d (taken to be 99% in this paper), neglecting nonalgorithmic overheads such as state-preparation and measurement time. In the adiabatic regime where $\ln[1-p_{\mathrm{GS}}(T)] \propto T\Delta_{\min}^2$ per the Landau-Zener formula [39], we have $\mathrm{TTS}_{\mathrm{QA}} \propto 1/\Delta_{\min}^2$, which is independent of T. However, it has been observed in some cases that it can be better to run QA nonadiabatically to obtain a shorter TTS [15–18]. By choosing the best annealing time T, regardless of adiabaticity, we can determine $\mathrm{TTS}_{\mathrm{QA}}^{\mathrm{opt}}$ as the minimum algorithmic run time of QA. For the QAOA, a similar metric can be defined. The variational parameters γ_i and β_i can be regarded as the time evolved under the Hamiltonians H_C and H_B, respectively. One can thus interpret the sum of the variational parameters to be the total "annealing" time, i.e., $T_p = \sum_{i=1}^{p}(|\gamma_i^*| + |\beta_i^*|)$ [45], and define

$$\mathrm{TTS}_{\mathrm{QAOA}}(p) = T_p\frac{\ln(1-p_d)}{\ln[1-p_{\mathrm{GS}}(p)]}, \tag{12}$$

$$\mathrm{TTS}_{\mathrm{QAOA}}^{\mathrm{opt}} = \min_{p>0} \mathrm{TTS}_{\mathrm{QAOA}}(p), \tag{13}$$

where $p_{\mathrm{GS}}(p)$ is the ground state population after the optimal p-level QAOA protocol. Note that this quantity does not take into account the overhead in finding the optimal parameters. We use $\mathrm{TTS}_{\mathrm{QAOA}}(p)$ here to benchmark the algorithm, and it should not be taken directly to be the actual experimental run time.

To compare the algorithms, we compute $\mathrm{TTS}_{\mathrm{QA}}^{\mathrm{opt}}$ and $\mathrm{TTS}_{\mathrm{QAOA}}^{\mathrm{opt}}$ for many random graph instances. For each even vertex number from $N = 10$ to $N = 18$, we randomly generate 1000 instances of w3R graphs. In Fig. 5(a), we plot the relationship between $\mathrm{TTS}_{\mathrm{QAOA}}^{\mathrm{opt}}$ and the minimum gap $\Delta_{\min}$ in quantum annealing for each instance. Most of the random graphs have large gaps ($\Delta_{\min} \gtrsim 10^{-2}$), and we observe that the optimal TTS follows the Landau-Zener prediction of $1/\Delta_{\min}^2$ for these graphs. This result indicates that a quasiadiabatic parametrization of the QAOA is the best when $\Delta_{\min}$ is reasonably large. Many graphs, however, exhibit very small gaps ($\Delta_{\min} \lesssim 10^{-3}$) and, thus, require an exceedingly long run time for adiabatic evolution. For some graphs, $\Delta_{\min}$ is as small as 10^{-8}, which implies that an adiabatic evolution requires a run time $T \gtrsim 10^{16}$. Nevertheless, we see that the QAOA can find the solution for these hard instances faster than adiabatic QA. Remarkably, $\mathrm{TTS}_{\mathrm{QAOA}}^{\mathrm{opt}}$ appears to be independent of the gap for all graphs that have extremely small gaps and beats the adiabatic TTS (Landau-Zener line) by many orders of magnitude. This result suggests that an exponential improvement of the TTS is possible with nonadiabatic mechanisms when adiabatic QA is limited by exponentially small gaps.

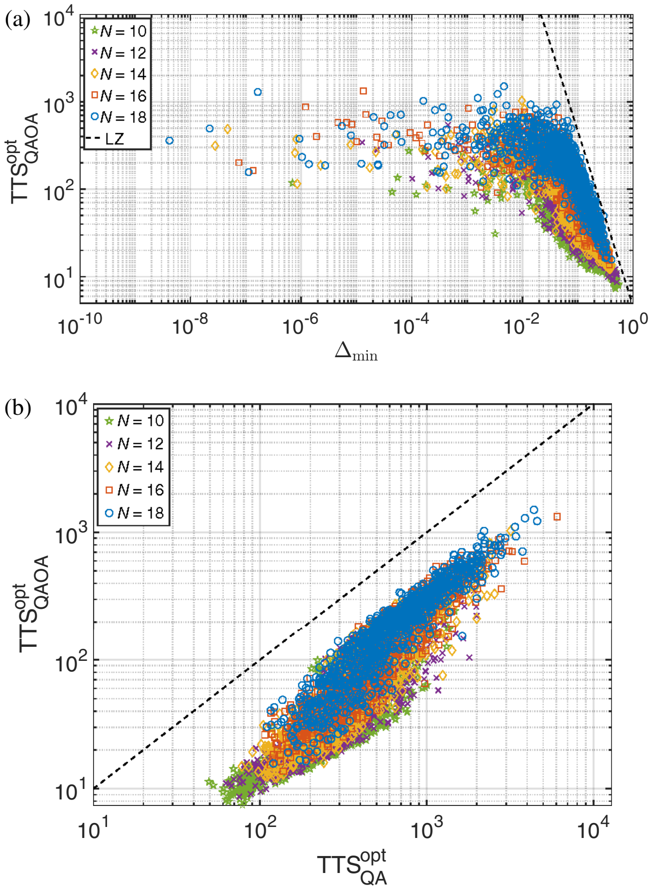

FIG. 5. (a) $\mathrm{TTS}_{\mathrm{QAOA}}^{\mathrm{opt}}$ versus the minimum spectral gap in QA, $\Delta_{\min}$, for many random w3R graph instances. 1000 random instances are generated for each graph vertex size N. The maximum cutoff p is taken to be $p_{\max} = 50, 50, 40, 35, 30$ for $N = 10$, 12, 14, 16, 18, respectively. The dashed line corresponds to the prediction of TTS $\propto 1/\Delta_{\min}^2$ in the adiabatic regime using the Landau-Zener formula [39]. (b) $\mathrm{TTS}_{\mathrm{QAOA}}^{\mathrm{opt}}$ versus $\mathrm{TTS}_{\mathrm{QA}}^{\mathrm{opt}}$ for each random graph instance. The dashed line indicates when the two are equal. The (Pearson's) correlation coefficient between the two is $\rho[\ln(\mathrm{TTS}_{\mathrm{QAOA}}^{\mathrm{opt}}), \ln(\mathrm{TTS}_{\mathrm{QA}}^{\mathrm{opt}})] \approx 0.91$.

Similarly for QA, the optimal annealing time T is often not in the adiabatic limit for small-gap graphs. In Fig. 5(b), we observe a strong correlation between $\mathrm{TTS}_{\mathrm{QAOA}}^{\mathrm{opt}}$ and $\mathrm{TTS}_{\mathrm{QA}}^{\mathrm{opt}}$ for each graph instance. This correlation suggests that QAOA is making use of the optimal annealing schedule, regardless of whether a slow adiabatic evolution or a fast diabatic evolution is better. We believe that if we use an optimized schedule beyond the linear ramp in Eq. (9), QA should be able to match the performance of the

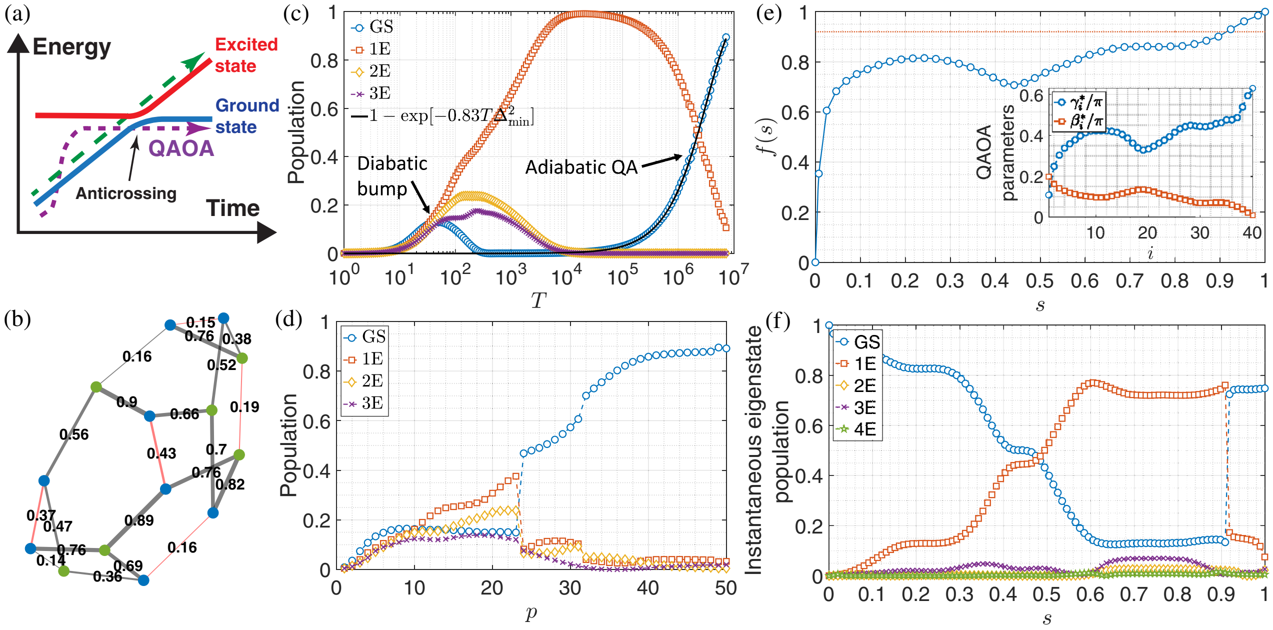

FIG. 6. (a) A schematic of how the QAOA and the interpolated annealing path can overcome the small minimum gap limitations via diabatic transitions (purple line). A naive adiabatic quantum annealing path leads to excited states passing through the anticrossing point (green line). (b) An instance of a weighted 3-regular graph that has a small minimum spectral gap along the quantum annealing path given by Eq. (9). The optimal MaxCut configuration is shown with two vertex colors, and the black (red) lines are the cut (uncut) edges. (c) Population in low-energy states after the quantum annealing protocol with different total times T. The black solid line follows the Landau-Zener formula for the ground state population, $p_{\mathrm{GS}} = 1 - \exp(-cT\Delta^2_{\min})$, where c is a fitting parameter. (d) Population in low-energy states using the QAOA at different levels p. The FOURIER heuristic strategy is used in the optimization. (e) Interpreting QAOA parameters (at $p = 40$) as a smooth quantum annealing path, via linear interpolation according to Eq. (14). The annealing time parameter $s = t/T_p$, where $T_p = \sum_{i=1}^{p}(|\gamma_i^*| + |\beta_i^*|)$. The red dotted line labels the location of anticrossing where the gap is at its minimum, at which point $f(s) \approx 0.92$. The inset shows the original QAOA optimal parameters γ_i^* and β_i^* for $p = 40$. (f) Instantaneous eigenstate population under the annealing path given in (e). Note that the instantaneous ground state and first excited state swap at the anticrossing point.

QAOA. In the following subsection, we take a representative example to explain our results observed in Fig. 5 and a mechanism of the QAOA to go beyond the adiabatic paradigm.

B. Beyond the adiabatic mechanism: A case study

To understand the behavior of the QAOA, we focus on graph instances that are hard for adiabatic QA in this subsection. In particular, we study a representative instance to explain how the QAOA as well as diabatic QA can overcome the adiabatic limitations. As illustrated in Fig. 6(a), we demonstrate that the QAOA can learn to utilize diabatic transitions at anticrossings to circumvent difficulties caused by very small gaps in QA.

Figure 6(b) shows a representative graph with $N = 14$, whose minimum spectral gap $\Delta_{\min} < 10^{-3}$. For this example hard instance, we first numerically simulate the quantum annealing process. Figure 6(c) shows populations in the ground state and low excited states at the end of the process for different annealing times T. Since the minimum gap is very small, the adiabatic condition requires $T \gtrsim 1/\Delta^2_{\min} \approx 10^6$. The Landau-Zener formula for the ground state population $p_{\mathrm{GS}} = 1 - \exp(-cT\Delta^2_{\min})$ fits well with our exact numerical simulation, where c is a fitting parameter. However, we can clearly observe a "bump" in the ground state population at annealing time $T \approx 40$. At such a timescale, the dynamics is clearly nonadiabatic; we call this phenomenon a *diabatic bump*. This phenomenon has been observed earlier in Refs. [15–18] for other optimization problems.

Subsequently, we simulate the QAOA on this hard instance. As mentioned earlier, although the QAOA optimizes energy instead of ground state overlap, a substantial ground state population can still be obtained even for many hard graphs. Using our FOURIER heuristic strategy, various low-energy state populations of the output state are shown for different levels p in Fig. 6(d). We observe that the QAOA can achieve a similar ground state population as the diabatic bump at small p and then substantially enhance it after $p \gtrsim 24$.

To better understand the mechanism of the QAOA and make a meaningful comparison with QA, we can interpret the QAOA parameters as a smooth annealing path. We again interpret the sum of the variational parameters to be the total annealing time, i.e., $T_p = \sum_{i=1}^{p}(|\gamma_i| + |\beta_i|)$. A smooth annealing path can be constructed from QAOA optimal parameters as

$$H_{\mathrm{QAOA}}(t) = -\{f(t)H_C + [1 - f(t)]H_B\},$$
$$f\left(t_i = \sum_{j=1}^{i}(|\gamma_j^*| + |\beta_j^*|) - \frac{1}{2}(|\gamma_i^*| + |\beta_i^*|)\right) = \frac{\gamma_i^*}{|\gamma_i^*| + |\beta_i^*|}, \tag{14}$$

where t_i is chosen to be at the midpoint of each time interval $(\gamma_i^* + \beta_i^*)$. With the boundary conditions $f(t = 0) = 0$ and $f(t = T_p) = 1$ and linear interpolation at other intermediate time t, we can convert QAOA parameters to a well-defined annealing path. We apply this conversion to the QAOA optimal parameters at $p = 40$ [Fig. 6(e)]. With this annealing path, we can follow the instantaneous eigenstate population throughout the quantum annealing process [Fig. 6(f)]. In contrast to adiabatic QA, the state population leaks out of the ground state and accumulates in the first excited state before the anticrossing point, where the gap is at its minimum. Using a diabatic transition at the anticrossing, the two states swap populations, and a large ground state population is obtained in the end. We note that the final state population from the constructed annealing path differs slightly from those of the QAOA, due to Trotterization and interpolation, but the underlying mechanism is the same, which is also responsible for the diabatic bump seen in Fig. 6(c). In addition to the conversion used in Eq. (14), we have tested a few other prescriptions to construct an annealing path from QAOA parameters, and qualitative features do not seem to change.

Hence, our results indicate that the QAOA is closely related to a cleverly optimized diabatic QA path that can overcome limitations set by the adiabatic theorem. Through optimization, the QAOA can find a good annealing path and exploit diabatic transitions to enhance ground state population. This result explains the observation in Fig. 5(a) that $\mathrm{TTS}_{\mathrm{QAOA}}^{\mathrm{opt}}$ can be significantly shorter than the time required by the adiabatic algorithm. On the other hand, as seen in Fig. 5(b), $\mathrm{TTS}_{\mathrm{QAOA}}^{\mathrm{opt}}$ is strongly correlated with $\mathrm{TTS}_{\mathrm{QA}}^{\mathrm{opt}}$: The QAOA automatically finds a good annealing path, which could be adiabatic or not, depending on what is the best route for the specific problem instance.

The effective dynamics of the QAOA for this specific instance, as we see in Fig. 6(f), can be understood mostly by an effective two-level system (see Appendix D for more discussions). In general, the energy spectrum can be more complex, and the dynamics may involve many excited states, which may not be explainable by the simple schematic in Fig. 6(a). Nonetheless, the QAOA can find a suitable path via our heuristic optimization strategies even in more complicated cases (see Appendix H and Ref. [46]).

VI. CONSIDERATIONS FOR EXPERIMENTAL IMPLEMENTATION

In this section, we discuss some important considerations for experimental realization. The framework of the QAOA is general and can be applied to various experimental platforms to solve combinatorial optimization problems. Here, we again focus on the MaxCut problem as a paradigmatic example, although it can also be applied to solve other interesting problems [46,47].

A. Finite measurement samples

So far, our focus is on understanding the best theoretically possible performance of the QAOA, assuming perfect measurement precision of the objective function in our numerical simulations. However, due to quantum fluctuations (i.e., projection noise) in actual experiments, the precision is finite, since it is obtained via averaging over finitely many measurement outcomes that can take on only discrete values. Hence, in practice, there is a trade-off between measurement cost and optimization quality: Finding a good optimum requires good precision at the cost of a large number of measurements [48]. Additionally, a large variance in the objective function value demands more measurements but may help improve the chances of finding near-optimal MaxCut configurations.

Here, we study the effect of measurement projection noise with a full Monte Carlo simulation of the QAOA on some example graphs, where the objective function is evaluated by repeated projective measurements until its error is below a threshold. More implementation details of this numerical simulation are discussed in Appendix F. In Fig. 7, we present the Monte Carlo simulation result for the example instance studied earlier in Fig. 6(b). Here, we simulate the QAOA by starting at either level $p = 1$ or $p = 5$ and increasing to higher p using our FOURIER heuristics after a local optimum is found at the current p. The initial parameters at $p = 1$ and $p = 5$, respectively, are chosen based on known optima found for smaller size instances. We see that the QAOA can find the MaxCut solution in approximately 10–10^2 measurements, and starting our optimization at an intermediate level ($p = 5$) is better than starting at the lowest level ($p = 1$). In comparison, random choices of initial parameters starting at $p = 1$ perform much worse, which fails to find the MaxCut solution until $10^3 - 10^4$ measurements are made. Moreover, when we compare the QAOA to QA with various annealing times, it appears that the choice of annealing time $T = 100$ can perform just as well as the QAOA in this instance. Nevertheless, running the QAOA at level $p = 5$ is still more advantageous than QA at $T = 100$, when coherence time is limited.

We remark that our simulation is limited only to small-size instances, and the good performance of the QAOA and QA we observe is complicated by the small but significant ground state population from generic annealing schedules.

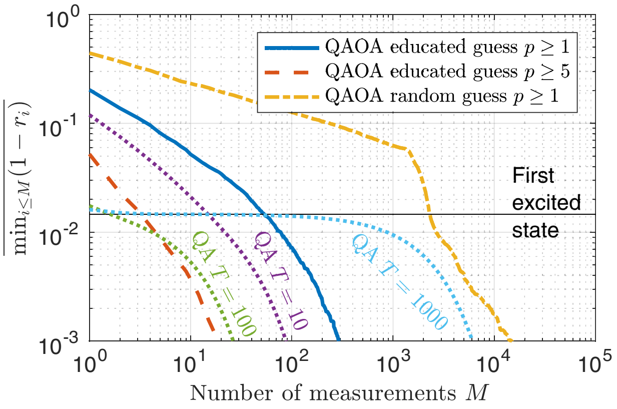

FIG. 7. Full Monte Carlo simulation of the QAOA accounting for measurement projection noise, on the example 14-vertex instance studied in Fig. 6(b). For various optimization strategies, we keep track of approximation ratio $r_i = |\mathrm{Cut}_i|/|\mathrm{MaxCut}|$ from the ith measurement and plot minimum fractional error $1 - r_i$ found after M measurements, averaged over many Monte Carlo realizations. The blue solid and red dashed lines correspond to QAOA optimized with the FOURIER strategy starting with an educated guess of $(\vec{\gamma}, \vec{\beta})$ at $p = 1$ and $p = 5$, respectively. The orange dash-dotted line corresponds to the QAOA optimized starting with random guesses at $p = 1$. The three labeled dotted lines are results from QA with total time $T = 10$, 10^2, 10^3.

Since it often takes 10^2 measurements to obtain a sufficiently precise estimate of the objective function, a ground state probability of $\gtrsim 10^{-2}$ means that one can find the ground state without much parameter optimization. Nevertheless, as quantum computers with $10^2 - 10^3$ qubits begin to come online, it will be interesting to see how the QAOA performs on much larger instances where the ground state probability from a generic QAOA or QA protocol is expected to be exponentially small, whereas the number of measurements necessary for optimization grows only polynomially with the problem size. The results here indicate that the parameter pattern and our heuristic strategies are practically useful guidelines in realistic implementations of the QAOA.

B. Implementation for large problem sizes

As experiments begin to test solving the MaxCut problems with quantum machines [6,8], limited quantum coherence time and graph connectivity are among the biggest challenges. In terms of coherence time, the QAOA is highly advantageous: The hybrid nature of the QAOA as well as its short- and intermediate-depth circuit parametrization makes it ideal for near-term quantum devices. In addition, our results show that the QAOA is not generally limited by small spectral gaps, which raises hopes to (approximately) solve interesting problems within the coherence time.

What would be the necessary problem size to explore a meaningful quantum advantage? We note that one of the leading exact classical solvers, the BiqMac solver [49], is able to solve MaxCut exactly up to $N \lesssim 100$ but takes a long time (more than an hour) for larger problem sizes. A fast heuristic algorithm, the breakout local search algorithm [50], can find the MaxCut solution with a high probability for a problem size of a few hundreds, although the solution is not guaranteed. Hence, a MaxCut problem with a few hundred vertices is an interesting regime to benchmark quantum algorithms. In terms of approximation, we noted earlier that the polynomial-time Goemans-Williamson algorithm has an approximation ratio guarantee of $r^* \approx 0.878$. It will be also interesting to find out if the QAOA is able to achieve a better approximation ratio for some problem instances where the exact solution is not obtainable, in this regime of hundreds of vertices.

We now discuss a few considerations that put these large-size problems in the experimentally feasible regime on near-term quantum systems.

1. *Reducing interaction range*

In a quantum experiment, each vertex can be represented by a qubit. For a large problem size, a major challenge to encode general graphs is the necessary range and versatility of the interaction patterns (between qubits). The embedding of a random graph into a physical implementation with a 1D or 2D geometry may require very long-range interactions. By relabeling the graph vertices, one can actually reduce the required range of interactions. This reduction can be formulated as the *graph bandwidth problem*: Given a graph $G = (V, E)$ with N vertices, a vertex numbering is a bijective map from vertices to distinct integers, $f: V \to \{1, 2, \ldots, N\}$. The bandwidth of a vertex numbering f is $B_f(G) = \max\{|f(u) - f(v)| : (u, v) \in E(G)\}$, which can be understood as the length of the longest edge (in 1D). The graph bandwidth problem is then to find the minimum bandwidth among all vertex numberings, i.e., $B(G) = \min_f B_f(G)$; namely, it is to minimize the length of the longest edge by vertex renumbering.

In general, finding the minimum graph bandwidth is NP-hard, but good heuristic algorithms have been developed to reduce the graph bandwidth. Figure 8(a) shows a simple example of bandwidth reduction. The top illustrates the vertex renumbering with a five-vertex graph. The bottom shows the histogram of graph bandwidths for 1000 random 3-regular graphs of $N = 400$ each. Using the Cuthill-McKee algorithm [51], the graph bandwidth can be reduced to around $B \approx 100$. While this reduction still requires quite a long interaction range in 1D, the bandwidth problem can also be generalized to higher dimensions. In 2D arrangements, we then expect the diameter of interaction to be $B_{2\mathrm{D}} \sim \sqrt{100}$ for $N = 400$ 3-regular graphs [52], which seems within reach for near-term quantum devices. A detailed study of the 2D bandwidth problem is beyond the scope of this work. Alternatively, one can make use of a general construction to encode any long-range interactions

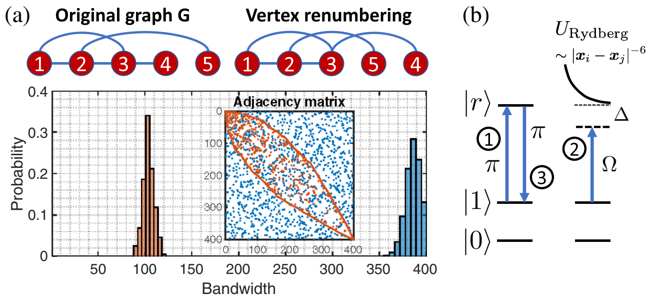

FIG. 8. (a) Vertex renumbering to reduce the graph bandwidth. Top: An example of vertex renumbering for a five-vertex graph. Bottom: Distribution of graph bandwidths for 1000 random 3-regular graphs with $N = 400$ each. The blue (orange) bars are the bandwidth before (after) vertex renumbering. The inset shows the sparsity pattern of the adjacency matrix before and after vertex renumbering for one particular graph. (b) A protocol to use a Rydberg-blockade controlled gate to implement the interaction term $\exp(-i\gamma\sigma_i^z\sigma_j^z)$. By choosing a proper gate time for the second step ($t = 2\pi/\sqrt{\Omega^2 + \Delta^2}$), one does not populate the Rydberg level $|r\rangle$. With tunable coupling strength Ω and detuning Δ, one can control the interaction time γ.

as local fields [53], although it requires additional physical qubits and gauge constraints. Apart from implementing arbitrary graphs in full generality, one can also restrict to special graphs that exhibit some geometric structures. For example, unit disk graphs are geometric graphs in the 2D plane, where vertices are connected by an edge only if they are within a unit distance. These graphs can be more naturally encoded into 2D physical implementations, and the MaxCut problem is still NP-hard on unit disk graphs [54].

2. Example implementation with Rydberg atoms

The above discussion is experimental-platform independent and is applicable to any state-of-the-art platforms, such as neutral Rydberg atoms [19,20,55–57], trapped ions [7,58–60], and superconducting qubits [5,6,61,62]. As an example, we briefly describe a feasible implementation of the QAOA with neutral atoms interacting via Rydberg excitations, where high-fidelity entanglement has been recently demonstrated [63]. We can use the hyperfine ground states in each atom to encode the qubit states $|0\rangle$ and $|1\rangle$, and the $|1\rangle$ state can be excited to the Rydberg level $|r\rangle$ to induce interactions. The qubit-rotating term $\exp(-i\beta\sum_{j=1}^{N}\sigma_j^x)$ can be implemented straightforwardly by a global driving beam with tunable durations. The interaction terms $\sum_{\langle i,j\rangle}\sigma_i^z\sigma_j^z$ can be implemented stroboscopically by Rydberg-blockade controlled gates [20], as illustrated in Fig. 8(b). By controlling the coupling strength Ω, detuning Δ, and the gate time, together with single-qubit rotations, one can implement $\exp(-i\gamma\sigma_i^z\sigma_j^z)$, which can then be repeated for each interacting pair. An additional major advantage of the Rydberg-blockade mechanism is its ability to perform multiqubit collective gates in parallel [20,64]. This ability can reduce the number of two-qubit operation steps from the number of edges to the number of vertices, N, which means a factor of N reduction for dense graphs with approximately N^2 edges. While the falloff of Rydberg interactions may limit the distance two qubits can interact, MaxCut problems of interesting sizes can still be implemented by vertex renumbering or focusing on unit disk graphs, as discussed above.

For generic problems of 400-vertex 3-regular graphs, we expect the necessary interaction range to be five atoms in 2D; assuming a minimum interatom separation of 2 μm, this range means an interaction radius of 10 μm, which is realizable with high Rydberg levels [20]. Given realistic estimates of coupling strength $\Omega \sim 2\pi \times 10$–$100$ MHz and single-qubit coherence time $\tau \sim 200$ μs (limited by Rydberg level lifetime), we expect with high-fidelity control that the error per two-qubit gate can be made roughly $(\Omega\tau)^{-1} \sim 10^{-3}$–$10^{-4}$. For 400-vertex 3-regular graphs, we can implement the QAOA of level $p \simeq \Omega\tau/N \sim 25$ with a 2D array of neutral atoms. We note that these are conservative estimates, since we do not consider advanced control techniques such as pulse shaping, and require less than one error in the entire quantum circuit; it is possible that the QAOA is not sensitive to some of the imperfections. Hence, we envision that soon the QAOA can be benchmarked against classical [25,49,50] and semiclassical [65–67] solvers on large problem sizes with near-term quantum devices.

VII. CONCLUSION AND OUTLOOK

In summary, we have studied the performance, nonadiabatic mechanism, and practical implementation of the QAOA applied to MaxCut problems. Our results provide important insights into the performance of the QAOA and suggest promising strategies for its practical realization on near-term quantum devices. Based on the observed patterns in the QAOA optimal parameters, we developed heuristic strategies for initializing optimization so as to efficiently find quasioptimal parameters in $O[\mathrm{poly}(p)]$ time. In contrast, optimization with the standard random initialization strategy that explores the entire parameter space needs $2^{O(p)}$ runs to obtain an equally good solution. We benchmarked, using these heuristic optimization strategies, the performance of the QAOA up to $p \leq 50$. On average, the performance characterized by the approximation ratio was observed to improve exponentially (or stretched exponentially) for random unweighted (or weighted) graphs. Focusing on hard graph instances where adiabatic QA fails due to extremely small spectral gaps, we found that the QAOA could learn via optimization a diabatic path to achieve significantly higher success probabilities to find the MaxCut solution. A metric taking into account both the success probability and the run time indicates the QAOA may not be limited by small spectral gaps. Finally, we provided a detailed resource analysis for experimental implementation of the QAOA on MaxCut and proposed a neutral-atom realization where problem sizes of a few hundred atoms are feasible in the near future.

As we have benchmarked via random MaxCut instances, our simple heuristic optimization strategies work very well. Nevertheless, more sophisticated methods could be developed to improve the performance and robustness. One possibility would be using machine-learning techniques to automatically learn and make use of the optimal parameter patterns to develop even more efficient parametrization and strategies (see, e.g., Ref. [68]). Another important but unsettled problem is assessing a reliable system-size scaling for the QAOA. On average, we observed a (stretched) exponential improvement with level p, up to $N = 22$. It remains open whether the same scaling will persist for a larger system size. For the hard instances we generated that have exceedingly small spectral gaps, the QAOA is able to overcome the adiabatic limitations in all cases examined; it remains to be seen how this behavior could extrapolate to larger problems. An experimental implementation with near-term quantum devices will be able to push the limit of our understanding and serve as a litmus test for genuine quantum speedup in solving practical problems.

Besides MaxCut, another interesting optimization problem is the maximum independent set (MIS) problem, which is also NP-hard and has many real-world applications [69,70]. In a separate work [46], we show that the MIS problem can be naturally encoded into the ground state of neutral atoms interacting via Rydberg excitations, with minimal overhead on the hardware. The QAOA would be a candidate quantum algorithm to solve the MIS optimization problem on the neutral-atom platform. The methodology we have developed here to efficiently optimize the QAOA on MaxCut can also be directly applied to the MIS problem, where we observe similar parameter patterns and nonadiabatic mechanisms of the QAOA; our results on MIS is discussed in more detail in Appendix H. With such rapid development in near-term quantum computers, we will soon be able to witness experimental tests of the capability of quantum algorithms to solve practically interesting problems.

ACKNOWLEDGMENTS

We thank Edward Farhi, Aram Harrow, Peter Zoller, Ignacio Cirac, Jeffrey Goldstone, Sam Gutmann, Wen Wei Ho, and Dries Sels for fruitful discussions. S.-T. W. in addition thanks Daniel Lidar, Matthias Troyer, and Johannes Otterbach for insightful discussions and suggestions during the Aspen Winter Conference on Advances in Quantum Algorithms and Computation. This work was supported through the National Science Foundation (NSF), the Center for Ultracold Atoms, the Air Force Office of Scientific Research via the MURI, the Vannevar Bush Faculty Fellowship, Department of Energy, the ONISQ program of the Defense Advanced Research Projects Agency, and Google Research Award. S. C. acknowledges support from the Miller Institute for Basic Research in Science. H. P. is supported by the NSF through a grant for the Institute for Theoretical Atomic, Molecular, and Optical Physics at Harvard University and the Smithsonian Astrophysical Observatory. Some of the computations in this paper were performed on the Odyssey cluster supported by the FAS Division of Science, Research Computing Group at Harvard University.

Note added.—Recently, we became aware of a related work appearing in Ref. [71]. In the preprint, they train the QAOA variational parameters on a batch of graph instances and compare the QAOA's performance with the classical Goemans-Williamson algorithm on small-size MaxCut problems. Similar parameter shapes are found, but Ref. [71] does not make use of any observed patterns to design optimization strategies. We, in addition, discover nonadiabatic mechanisms of the QAOA, which is not studied in Ref. [71].

APPENDIX A: OPTIMAL PARAMETER PATTERN FOR WEIGHTED GRAPHS

Here, we illustrate the pattern of optimal QAOA parameters for both weighted 3-regular (w3R) graphs and weighted complete graphs. The weight of each edge is randomly drawn from uniform distribution on the interval [0,1]. In Fig. 9, we illustrate the pattern by simultaneously plotting the optimal parameters for 40 instances. In both cases, we see a pattern analogous to what is found for unweighted 3-regular (u3R) graphs in Fig. 2(b), where γ_i^* tend to increase smoothly and β_i^* tend to decrease smoothly with $i = 1, 2, \ldots, p$. The optimal parameter of graphs from the same class also appears to cluster together in the same range.

We observe that the spread of $\vec{\gamma}^*$ for w3R graphs in Fig. 9(a) is wider than that for u3R graphs in Fig. 2(b). This difference is because the random weights effectively increase the number of subgraph types. Moreover, the larger value for $\vec{\gamma}^*$ for w3R compared to u3R graphs can be understood via the effective mean-field strength that each qubit experiences.

For complete graphs in Fig. 9(b), we observe that $\vec{\gamma}^*$ for different weighted graphs have a narrower spread as well as a smaller value compared to both u3R and w3R graphs. This difference is because there is only one type of subgraph that every edge sees. The nonzero spread is attributed to the fact that there are random weights on the edges. We also expect this spread to vanish as problem size N increases, when, for a typical instance, the distribution of weights on the $N - 1$ edges incident to each qubits converges. The smaller value of $\vec{\gamma}^*$ can also be understood via the effective mean-field strength picture, as each qubit interacts with all $N - 1$ other qubits.

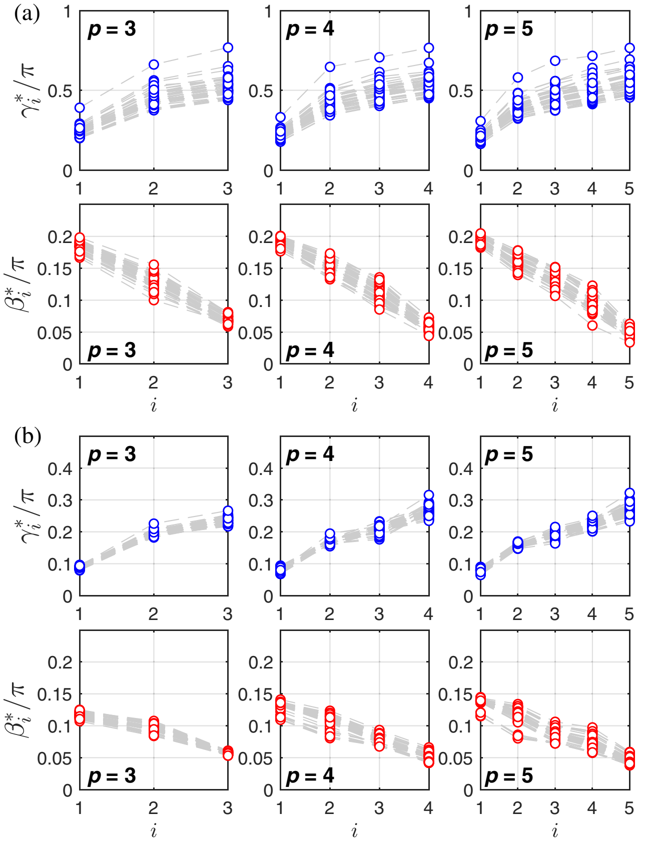

FIG. 9. Optimal QAOA parameters $(\vec{\gamma}^*, \vec{\beta}^*)$ for 40 instances of 16-vertex (a) w3R graphs and (b) weighted complete graphs, for $3 \le p \le 5$. Each dashed line connects parameters for one particular graph instance. For each instance and each p, the BFGS routine is used to optimize from 10^4 random initial points, and the best parameters are kept as $(\vec{\gamma}^*, \vec{\beta}^*)$.

APPENDIX B: DETAILS OF HEURISTIC OPTIMIZATION STRATEGIES

In the main text, we propose two classes of heuristic strategies called INTERP and FOURIER for generating initial points in optimizing QAOA parameters. Both INTERP and FOURIER strategies work well for all the instances we examine. We choose to use FOURIER for the results in the main text due to the slight edge in its performance in finding better optima when random perturbations are introduced. We now explain how these strategies work in detail and compare their performances.

1. Interpolation-based strategy

In the optimization strategy that we call INTERP, we use linear interpolation to produce a good initial point for optimizing the QAOA as one iteratively increases the level p. This strategy is based on the observation that the shape of parameters $(\vec{\gamma}^*_{(p+1)}, \vec{\beta}^*_{(p+1)})$ closely resembles that of $(\vec{\gamma}^*_{(p)}, \vec{\beta}^*_{(p)})$.

The strategy works as follows: For a given instance, we iteratively optimize the QAOA starting from $p = 1$ and increment p after obtaining a local optimum $(\vec{\gamma}^L_{(p)}, \vec{\beta}^L_{(p)})$. In order to optimize parameters for level $p+1$, we take the optimized parameters from level p and produce initial points $(\vec{\gamma}^0_{(p+1)}, \vec{\beta}^0_{(p+1)})$ according to

$$[\vec{\gamma}^0_{(p+1)}]_i = \frac{i-1}{p}[\vec{\gamma}^L_{(p)}]_{i-1} + \frac{p-i+1}{p}[\vec{\gamma}^L_{(p)}]_i \qquad \text{(B1)}$$

for $i = 1, 2, \ldots, p+1$. Here, we denote $[\vec{\gamma}]_i \equiv \gamma_i$ as the ith element of the parameter vector $\vec{\gamma}$, and $[\vec{\gamma}^L_{(p)}]_0 \equiv [\vec{\gamma}^L_{(p)}]_{p+1} \equiv 0$. The expression for $\vec{\beta}^0_{(p+1)}$ is the same as above after replacing $\gamma \to \beta$. Starting from $(\vec{\gamma}^0_{(p+1)}, \vec{\beta}^0_{(p+1)})$, we then optimize (e.g., using the BFGS routine) to obtain a local optimum $(\vec{\gamma}^L_{(p+1)}, \vec{\beta}^L_{(p+1)})$ for the $(p+1)$-level QAOA. Finally, we increment p by one and repeat the same process until a target level is reached.

2. Details of FOURIER[q, R] strategy

We now provide the details of our second heuristic strategy for optimizing the QAOA that we call FOURIER. As described in the main text (Sec. III B), here we use a new representation of the p-level QAOA parameters as $(\vec{u}, \vec{v}) \in \mathbb{R}^{2q}$, where

$$\gamma_i = \sum_{k=1}^{q} u_k \sin\left[\left(k - \frac{1}{2}\right)\left(i - \frac{1}{2}\right)\frac{\pi}{p}\right],$$
$$\beta_i = \sum_{k=1}^{q} v_k \cos\left[\left(k - \frac{1}{2}\right)\left(i - \frac{1}{2}\right)\frac{\pi}{p}\right]. \qquad \text{(B2)}$$

Roughly speaking, the FOURIER strategy works by starting with level $p = 1$, optimizing, and then reusing the optimum at level p in $(\vec{u}, \vec{v})$ representation to generate a good initial point for level $p+1$.

In fact, we propose several variants of the FOURIER strategy for optimizing p-level QAOA: They are called FOURIER[q, R], characterized by two integer parameters q and R. The first integer q labels the maximum frequency component we allow in the amplitude parameters $(\vec{u}, \vec{v})$. When we set $q = p$, the full power of the p-level QAOA can be utilized, in which case we simply denote the strategy as FOURIER[∞, R], since q grows unbounded with p. Nevertheless, the smoothness of the optimal parameters $(\vec{\gamma}, \vec{\beta})$ implies that only the low-frequency components are important. Thus, we can also consider the case where q is a fixed constant independent of p, so the number of parameters is bounded even as the QAOA circuit depth increases. The second integer R is the number of random perturbations we add to the parameters so that we can sometimes escape a local optimum toward a better one. For the results shown in the main text, we use the FOURIER$[\infty, 10]$ strategy, meaning we set $q = p$ and $R = 10$, unless otherwise stated.

In the basic FOURIER$[\infty, 0]$ variant of this strategy, we generate a good initial point for level $p + 1$ by adding a higher-frequency component, initialized at zero amplitude, to the optimum at level p. More concretely, we take parameters from a local optimum $(\vec{\boldsymbol{u}}^L_{(p)}, \vec{\boldsymbol{v}}^L_{(p)}) \in \mathbb{R}^{2p}$ at level p and generate an initial point $(\vec{\boldsymbol{u}}^0_{(p+1)}, \vec{\boldsymbol{v}}^0_{(p+1)}) \in \mathbb{R}^{2(p+1)}$ according to

$$\vec{\boldsymbol{u}}^0_{(p+1)} = (\vec{\boldsymbol{u}}^L_{(p)}, 0), \qquad \vec{\boldsymbol{v}}^0_{(p+1)} = (\vec{\boldsymbol{v}}^L_{(p)}, 0). \tag{B3}$$

Using $(\vec{\boldsymbol{u}}^0_{(p+1)}, \vec{\boldsymbol{v}}^0_{(p+1)})$ as an initial point, we perform the BFGS optimization routine to obtain a local optimum $(\vec{\boldsymbol{u}}^L_{(p+1)}, \vec{\boldsymbol{v}}^L_{(p+1)})$ for the level $p + 1$.

We also consider an improved variant of the strategy, FOURIER$[\infty, R > 0]$, which is sketched alongside the $R = 0$ variant in Fig. 10. This variant is motivated by our observation that the basic FOURIER$[\infty, 0]$ strategy can sometimes get stuck at a suboptimal local optimum, whereas perturbing its initial point can improve the performance of the QAOA. Here, in addition to optimizing according to the basic strategy, we optimize the $(p + 1)$-level QAOA from $R + 1$ extra initial points, R of which are generated by adding random perturbations to the best of all local optima $(\vec{\boldsymbol{u}}^B_{(p)}, \vec{\boldsymbol{v}}^B_{(p)})$ found at level p. Specifically, for each instance at the $(p + 1)$-level QAOA, and for $r = 0, 1, \ldots, R$, we optimize starting from

$$\vec{\boldsymbol{u}}^{0,r}_{(p+1)} = \begin{cases} (\vec{\boldsymbol{u}}^B_{(p)}, 0), & r = 0, \\ (\vec{\boldsymbol{u}}^B_{(p)} + \alpha \vec{\boldsymbol{u}}^{P,r}_{(p)}, 0), & 1 \le r \le R, \end{cases}$$
$$\vec{\boldsymbol{v}}^{0,r}_{(p+1)} = \begin{cases} (\vec{\boldsymbol{v}}^B_{(p)}, 0), & r = 0, \\ (\vec{\boldsymbol{v}}^B_{(p)} + \alpha \vec{\boldsymbol{v}}^{P,r}_{(p)}, 0), & 1 \le r \le R, \end{cases} \tag{B4}$$

where $\vec{\boldsymbol{u}}^{P,r}_{(p)}$ and $\vec{\boldsymbol{v}}^{P,r}_{(p)}$ contain random numbers drawn from normal distributions with mean 0 and variance given by $\vec{\boldsymbol{u}}^B_{(p)}$ and $\vec{\boldsymbol{v}}^B_{(p)}$:

$$[\vec{\boldsymbol{u}}^{P,r}_{(p)}]_k \sim \text{Normal}(0, [\vec{\boldsymbol{u}}^B_{(p)}]^2_k),$$
$$[\vec{\boldsymbol{v}}^{P,r}_{(p)}]_k \sim \text{Normal}(0, [\vec{\boldsymbol{v}}^B_{(p)}]^2_k). \tag{B5}$$

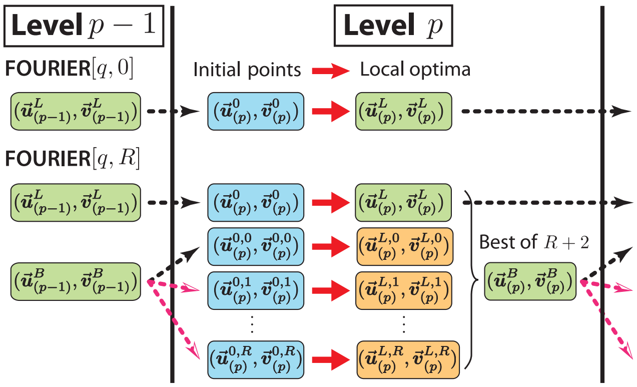
Level $p - 1$
Level p
FOURIER$[q, 0]$
Initial points
Local optima
$(\vec{\boldsymbol{u}}^L_{(p-1)}, \vec{\boldsymbol{v}}^L_{(p-1)})$
$(\vec{\boldsymbol{u}}^0_{(p)}, \vec{\boldsymbol{v}}^0_{(p)})$
$(\vec{\boldsymbol{u}}^L_{(p)}, \vec{\boldsymbol{v}}^L_{(p)})$
FOURIER$[q, R]$
$(\vec{\boldsymbol{u}}^L_{(p-1)}, \vec{\boldsymbol{v}}^L_{(p-1)})$
$(\vec{\boldsymbol{u}}^0_{(p)}, \vec{\boldsymbol{v}}^0_{(p)})$
$(\vec{\boldsymbol{u}}^L_{(p)}, \vec{\boldsymbol{v}}^L_{(p)})$
$(\vec{\boldsymbol{u}}^{0,0}_{(p)}, \vec{\boldsymbol{v}}^{0,0}_{(p)})$
$(\vec{\boldsymbol{u}}^{L,0}_{(p)}, \vec{\boldsymbol{v}}^{L,0}_{(p)})$
Best of $R + 2$
$(\vec{\boldsymbol{u}}^B_{(p-1)}, \vec{\boldsymbol{v}}^B_{(p-1)})$
$(\vec{\boldsymbol{u}}^{0,1}_{(p)}, \vec{\boldsymbol{v}}^{0,1}_{(p)})$
$(\vec{\boldsymbol{u}}^{L,1}_{(p)}, \vec{\boldsymbol{v}}^{L,1}_{(p)})$
$(\vec{\boldsymbol{u}}^B_{(p)}, \vec{\boldsymbol{v}}^B_{(p)})$
$(\vec{\boldsymbol{u}}^{0,R}_{(p)}, \vec{\boldsymbol{v}}^{0,R}_{(p)})$
$(\vec{\boldsymbol{u}}^{L,R}_{(p)}, \vec{\boldsymbol{v}}^{L,R}_{(p)})$

FIG. 10. Schematics of the two variants of the FOURIER$[q, R]$ heuristic strategy for optimizing the QAOA, when $R = 0$ and $R > 0$. Optimized parameters (green) at level $p - 1$ are used to generate good initial points (blue) for optimizing at level p; the same process is repeated to optimize for level $p + 1$. When generating initial points, black dashed arrows indicate appending a higher-frequency component [Eq. (B3)], and pink dashed arrows correspond to adding random perturbations [Eq. (B4)]. In the $R > 0$ variant, two local optima (green) are kept in parallel at each level p for generating good initial points: $(\vec{\boldsymbol{u}}^L_{(p)}, \vec{\boldsymbol{v}}^L_{(p)})$ is the same optimum obtained from the FOURIER$[q, 0]$ strategy, and $(\vec{\boldsymbol{u}}^B_{(p)}, \vec{\boldsymbol{v}}^B_{(p)})$ is the best of $R + 2$ local optima, R of which are obtained from perturbed initial points. We find that keeping the $(\vec{\boldsymbol{u}}^L_{(p)}, \vec{\boldsymbol{v}}^L_{(p)})$ optimum improves the stability of the heuristic, as random perturbations can sometimes lead to erratic and nonsmooth optimized parameters.

There is a free parameter α corresponding to the strength of the perturbation. Based on our experience from trial and error, setting $\alpha = 0.6$ has consistently yielded good results. This choice of α is assumed for the results in this paper.

We remark that, while the INTERP strategy can also get stuck in a local optimum, we find that adding perturbations to INTERP does not work as well as to FOURIER. We attribute this difference to the fact that the optimal parameters are smooth, and adding perturbations in the $(\vec{\boldsymbol{u}}, \vec{\boldsymbol{v}})$ space modifies $(\vec{\boldsymbol{\gamma}}, \vec{\boldsymbol{\beta}})$ in a correlated way, which could enable the optimization to escape local optima more easily. Hence, we consider here FOURIER with perturbations but not INTERP.

Finally, we also consider variants of the FOURIER strategy where the number of frequency components q is fixed. These variants are the same as aforementioned strategies where $q = p$, except we truncate each of the $\vec{\boldsymbol{u}}$ and $\vec{\boldsymbol{v}}$ parameters to keep only the first q components. For example, when optimizing the QAOA at level $p \geq q$ with the FOURIER$[q, 0]$ strategy, we stop adding higher-frequency components and use the initial point $\vec{\boldsymbol{u}}^0_{(p+1)} = \vec{\boldsymbol{u}}^L_{(p)} \in \mathbb{R}^q$.

3. Comparison between heuristics

We now examine the difference in the performance among the various heuristic strategies we propose. For our example instance in Fig. 11, we note that the INTERP and FOURIER$[\infty, 0]$ strategies have essentially identical performance (except for small variations barely visible at large p). This result is expected, since both strategies generate initial points for optimizing level $p + 1$ based on smooth deformation of the optimum at level p. In any case, the FOURIER$[\infty, 10]$ strategy outperforms INTERP and

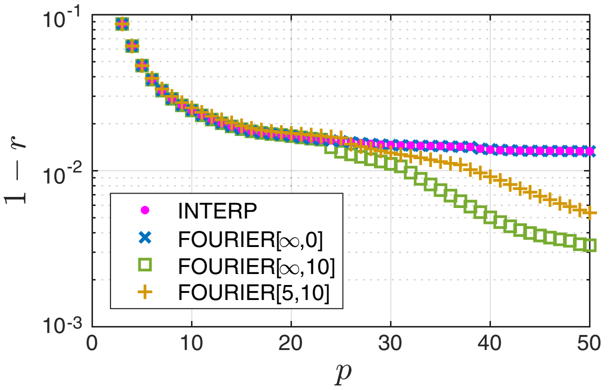

FIG. 11. The fractional error achieved by optimizing using our various heuristics on an example instance of a 14-vertex w3R graph. This instance is the same shown in Fig. 6(b).

FOURIER$[\infty,0]$ beginning at level $p \gtrsim 20$. Interestingly, even when restricting the number of QAOA parameters to $2q = 10$, the FOURIER[5,10] strategy closely matches the performance of other heuristics at low p and beats the $R = 0$ heuristic at large p. This result suggests that the optimal QAOA parameters may, in fact, live in a small-dimensional manifold. Therefore, optimization for the intermediate p-level QAOA can be dramatically simplified by using our new parameterization $(\vec{u}, \vec{v})$.

APPENDIX C: TTS FOR EXAMPLE GRAPH INSTANCE

In Sec. V B, we focus on a representative graph instance where the adiabatic minimum gap is small [Fig. 6(b)]. The low-energy spectrum for the graph along the QA path can be seen in Fig. 12(a). We remark that only eigenstates that are invariant under the parity operator $\mathcal{P} = \prod_{i=1}^{N} \sigma_i^x$ are shown, since the Hamiltonian $H_{\mathrm{QA}}(s)$ commutes with $\mathcal{P}$ and the initial state is $\mathcal{P}$ invariant: $\mathcal{P}|\psi(0)\rangle = |\psi(0)\rangle$.

Figures 12(b) and 12(c) illustrate $\mathrm{TTS}_{\mathrm{QA}}$ and $\mathrm{TTS}_{\mathrm{QAOA}}$ for the same graph. For QA, one can see that nonadiabatic evolution with $T \approx 20$ yields orders of magnitude shorter TTS than the adiabatic evolution. We also see that the TTS in the adiabatic limit is independent of the annealing time T, following the Landau-Zener formula $p_{\mathrm{GS}} = 1 - \exp(-cT\Delta_{\min}^2)$. For the QAOA, we use our FOURIER $[\infty, 10]$ heuristic strategy to perform the numerical simulation up to $p_{\max} = 50$ and compute $\mathrm{TTS}_{\mathrm{QAOA}}(p)$ and $\mathrm{TTS}_{\mathrm{QAOA}}^{\mathrm{opt}}$ (up to $p \le p_{\max}$). For this particular graph, we note that, although $\mathrm{TTS}_{\mathrm{QAOA}}^{\mathrm{opt}}$ occurs at $p = 49$, in practice it may be better to run the QAOA at $p = 4$ or $p = 5$ due to optimization overhead and error accumulation at deeper circuit depths. The apparent discontinuous jump in $\mathrm{TTS}_{\mathrm{QAOA}}$ is due to the corresponding jump in $p_{\mathrm{GS}}(p)$, which can be explained by two reasons: First, our heuristic strategy is not guaranteed to find the global optimum, and random perturbations may help the algorithm escape a local optimum, resulting in a jump in the ground state population; second, even when the global optimum is found for all level p, there can still be discontinuities in p_{GS}, since the objective function of the QAOA is energy instead of ground state population.

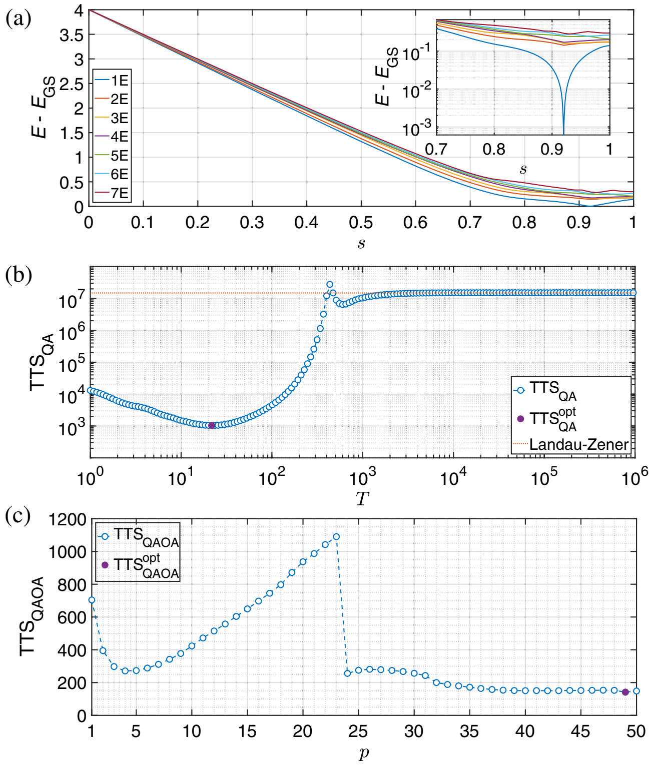

FIG. 12. (a) Energy spectrum of low excited states (measured from the ground state energy E_{GS}) along the annealing path for the graph instance given in Fig. 6(b). Only states that can couple to the ground state are shown, i.e., states that are invariant under the parity operator $\mathcal{P} = \prod_{i=1}^{N} \sigma_i^x$. The inset shows the energy gap from the ground state in the logarithmic scale. (b) Time to solution for the linear-ramp quantum annealing protocol, $\mathrm{TTS}_{\mathrm{QA}}$, for the same graph instance. The TTS in the long time limit follows a line predicted by the Landau-Zener formula, which is independent of the annealing time T. (c) The TTS for the QAOA at each iteration depth p.

APPENDIX D: EFFECTIVE FEW-LEVEL UNDERSTANDING OF THE DIABATIC BUMP

In this Appendix, we elucidate the mechanism of the diabatic bump observed in Fig. 6(c) via an effective few-level dynamics. To study the intermediate dynamics during quantum annealing, we can work in the basis of instantaneous eigenstates $|\epsilon_l(t)\rangle$, where

$$H_{\mathrm{QA}}(t)|\epsilon_l(t)\rangle = \epsilon_l(t)|\epsilon_l(t)\rangle. \tag{D1}$$

Expanding the time-evolved state in this basis as $|\psi(t)\rangle = \sum_l a_l(t)|\epsilon_l(t)\rangle$, the Schrödinger equation can be written as

$$i\sum_l (\dot{a}_l|\epsilon_l\rangle + a_l|\dot{\epsilon}_l\rangle) = \sum_l \epsilon_l a_l |\epsilon_l\rangle, \tag{D2}$$

where $\hbar = 1$ and the time dependence in the notations is dropped for convenience. Multiplying the equation by $\langle\epsilon_k|$, the Schrödinger equation becomes

$$i\dot{a}_k = \epsilon_k a_k - i\sum_{l\neq k} a_l \langle\epsilon_k|\dot{\epsilon}_l\rangle, \tag{D3}$$

where $\langle\epsilon_k|\dot{\epsilon}_k\rangle = 0$ is taken by absorbing the phase into the eigenvector $|\epsilon_k\rangle$. Written in a matrix form, we have

$$i\begin{pmatrix}\dot{a}_0\\ \dot{a}_1\\ \dot{a}_2\\ \vdots\end{pmatrix} = \begin{pmatrix} 0 & -i\langle\epsilon_0|\dot{\epsilon}_1\rangle & -i\langle\epsilon_0|\dot{\epsilon}_2\rangle & \cdots\\ -i\langle\epsilon_1|\dot{\epsilon}_0\rangle & \Delta_{10} & -i\langle\epsilon_1|\dot{\epsilon}_2\rangle & \cdots\\ -i\langle\epsilon_2|\dot{\epsilon}_0\rangle & -i\langle\epsilon_2|\dot{\epsilon}_1\rangle & \Delta_{20} & \cdots\\ \vdots & \vdots & \vdots & \ddots\end{pmatrix}\begin{pmatrix}a_0\\ a_1\\ a_2\\ \vdots\end{pmatrix}, \tag{D4}$$

where we take the ground state energy $\epsilon_0 = 0$ (by absorbing it into the phase of the coefficients) and $\Delta_{i0} = \epsilon_i - \epsilon_0$ is the instantaneous energy gap from the ith excited state to the ground state. The time evolution starts from the initial ground state with $a_0 = 1$ and $a_i = 0$ for $i \neq 0$, and the adiabatic condition to prevent coupling to excited states is

$$\frac{|\langle\epsilon_0|\dot{\epsilon}_i\rangle|}{\Delta_{i0}} = \frac{|\langle\epsilon_0|\frac{dH_{\mathrm{QA}}}{dt}|\epsilon_i\rangle|}{\Delta_{i0}^2} = \frac{|\langle\epsilon_0|\frac{dH_{\mathrm{QA}}}{ds}|\epsilon_i\rangle|}{\Delta_{i0}^2 T} \ll 1. \tag{D5}$$

The first equality can be derived from Eq. (D1). This condition gives the standard adiabatic timescale $T = O(1/\Delta_{\min}^2)$. As we discuss in Sec. V B, the minimum gap for some graphs can be exceedingly small, so the adiabatic limit is not practical. However, it may be possible to choose an appropriate run time T which breaks adiabaticity but is long enough such that only a few excited states are effectively involved in the dynamics. This regime is where the diabatic bump operates, and one can understand the dynamics by truncating Eq. (D4) to the first few basis states.

As an example, we plot in Fig. 13(a) the instantaneous eigenstate populations of the first few states. It is simulated with the full Hilbert space, but effectively the same dynamics will be generated if the simulation is restricted to the first few basis states in Eq. (D4). Figure 13(b) shows the strength of the couplings between the instantaneous ground state and the low excited states. By comparing Figs. 13(a) and 13(b), one can see that $T = T^* = 40$ allows the time evolution to break the adiabatic condition before the anticrossing: Population leaks to the first excited state, which becomes the ground state after the anticrossing. Thus, the timescale of T^* for the diabatic bump represents a delicate balance between allowing population to leak out of the ground state and suppressing excessive population leakage, which explains why it happens at a certain range of timescale.

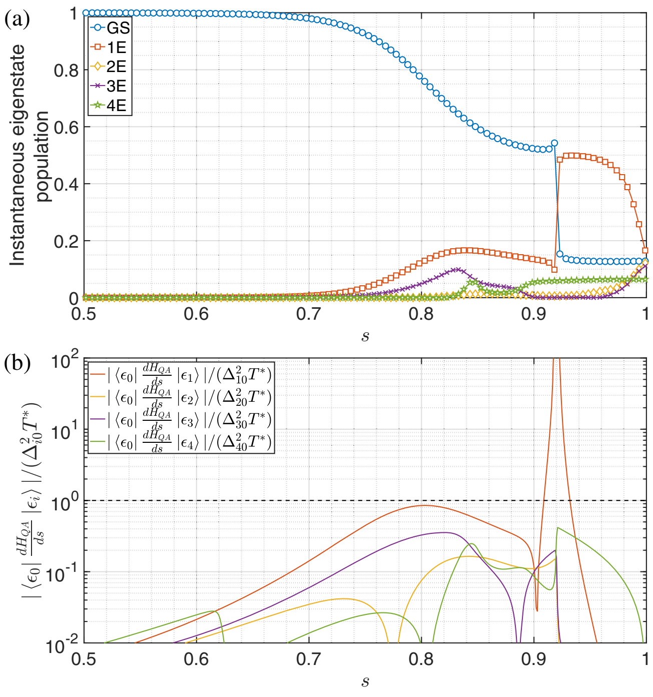

FIG. 13. (a) Instantaneous eigenstate populations along the linear-ramp quantum annealing path given by Eq. (9) for the example graph in Fig. 6(b). The annealing time is chosen to be $T = T^* = 40$, which corresponds to the time where the diabatic bump occurs in Fig. 6(c). (b) Coupling between the instantaneous ground states and the first few excited states. The plotted quantities measure the degree of adiabaticity [as explained in Eq. (D5)].

APPENDIX E: COMPARING DIFFERENT CLASSICAL OPTIMIZATION ROUTINES

In this Appendix, we compare three different classical routines that can be used to optimize QAOA parameters: Bayesian optimization [38], Nelder-Mead [37], and BFGS [35]. This comparison is done by a numerical experiment where we apply these optimization routines to ten instances of 14-vertex w3R graphs. To compare them on equal footing, we terminate each optimization run after a budget of $20p$ objective function evaluations is used. In the gradient-based routine BFGS, we include the cost of gradient estimation via the finite-difference method into the budget of $20p$ objective function evaluations. For each routine, we start at $p = 1$, gradually increment p, and perform optimization where the initial point is generated using either our heuristic strategies (FOURIER and INTERP) or the standard strategy of RI. We use the versions of these optimization routines implemented in MATLAB R2017b as bayesopt, fminsearch, and fminunc, respectively. The objective function at each set of parameters is evaluated to floating point precision. The tolerance

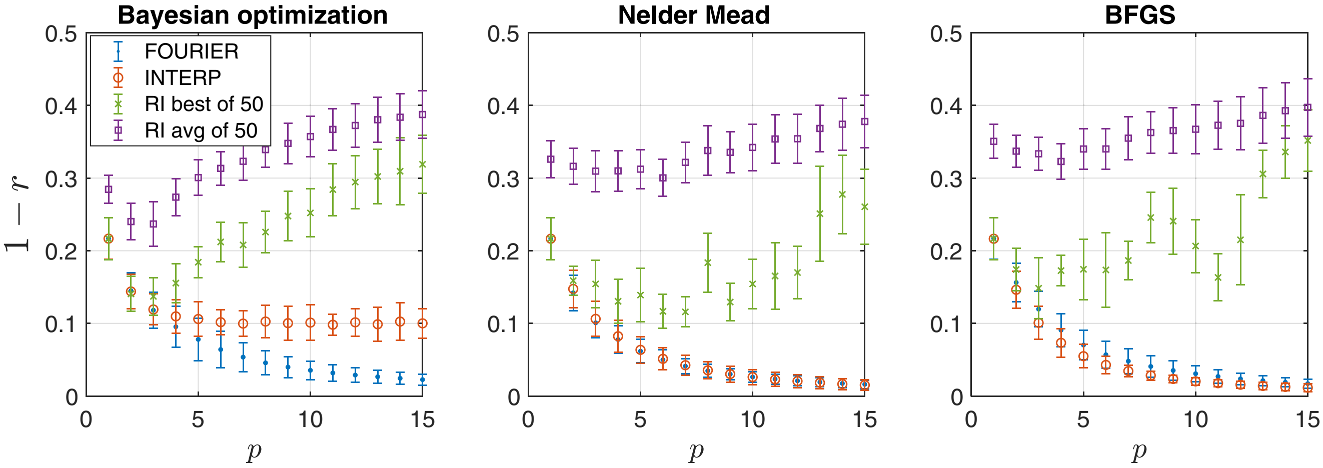

FIG. 14. Comparison of three different optimization routines applied to ten instances of 14-vertex w3R graphs. The initial point of optimization is generated with either our heuristics (FOURIER or INTERP) or random initialization (RI). We plot the fractional error $1-r$, averaged over instances, for each optimization routine and each initialization strategy at each p. The error bars are sample standard deviations from the ten instances. For our heuristic strategy, the optimization starts with a single initial point generated using our FOURIER[∞, 0] or INTERP heuristics as described in Appendix B. For the RI strategy, we generate 50 random initial points uniformly in the parameter space and optimize from each initial point; both the best and the average of 50 RI runs are plotted.

in both the objective function value and step size in parameter space, as well as the finite-difference-gradient step size, is chosen to be 0.01.

The result of our numerical experiment is plotted in Fig. 14. Similar to Fig. 3(a), we see that the average quality of local optimum found from 50 RI runs is much worse than the best, indicating the difficulty of optimizing in the QAOA parameter landscape without a good initial point. We also see that, regardless of the classical routines chosen, one run of optimization from an initial point generated from our heuristic strategies is generally better than the best out of 50 runs from randomly generated initial points. This result indicates that our heuristic strategies work much better than RI and can be integrated with a variety of classical optimization routines. Moreover, we find that Bayesian optimization typically does not do as well as Nelder-Mead or BFGS for larger p, which is consistent with the folklore that this routine is better suited for low-dimensional parameter space [38]. On the other hand, it seems both Nelder-Mead and BFGS have comparable performance, even when the cost of gradient evaluation is taken into account. The slight difference we observe between the two in Fig. 14 is inconclusive and can be attributed to suboptimal choices of tolerance and step size parameters and the deliberately limited budget of objective function evaluations.

APPENDIX F: DETAILS OF SIMULATION WITH MEASUREMENT PROJECTION NOISE

When running the QAOA on actual quantum devices, the objective function is evaluated by averaging over many measurement outcomes, and, consequently, its precision is limited by the so-called measurement projection noise from quantum fluctuations. We account for this effect by performing full Monte Carlo simulations of actual measurements, where the simulated quantum processor outputs only approximate values of the objective function obtained by averaging M measurements:

$$\bar{F}_{p,M} = \frac{1}{M}\sum_{i=1}^{M} C(z_{p,i}), \tag{F1}$$

where $z_{p,i}$ is a random variable corresponding to the ith measurement outcome obtained by measuring $|\psi_p(\vec{\gamma},\vec{\beta})\rangle$ in the computational basis and $C(z)$ is the classical objective function. Note that, when $M\to\infty$, we obtain $\bar{F}_{p,M}\to F_p = \langle\psi_p(\vec{\gamma},\vec{\beta})|H_C|\psi_p(\vec{\gamma},\vec{\beta})\rangle$. In the simulation, we achieve finite precision $|\bar{F}_{p,M}-F_p|\lesssim\xi$ by sampling measurements until the cumulative standard error of the mean falls below the target precision level ξ. In other words, for each evaluation of F_p requested by the classical optimizer, the number of measurements M performed is set by the following criterion:

$$\sqrt{\frac{1}{M(M-1)}\sum_{i=1}^{M}[C(z_{p,i})-\bar{F}_{p,M}]^2} \le \xi. \tag{F2}$$

Roughly, we expect $M\approx \mathrm{Var}(H_C)/\xi^2$. To address issues with finite sample sizes, we also require that at least ten measurements be performed ($M\ge 10$) for each objective function evaluation.

We now provide some details on the classical optimization algorithm used to optimize QAOA parameters. Generally, classical optimization algorithms iteratively uses information from some given parameter point $(\vec{\gamma},\vec{\beta})$ to find a new parameter point $(\vec{\gamma}',\vec{\beta}')$ that hopefully produces a larger value of the objective function $F_p(\vec{\gamma}',\vec{\beta}') \geq F_p(\vec{\gamma},\vec{\beta})$. In order for the algorithm to terminate, we need to set some stopping criteria. Here, we specify two: First, we set an objective function tolerance ϵ, such that if the change in objective function $|\bar{F}_{p,M'}(\vec{\gamma}',\vec{\beta}') - \bar{F}_{p,M}(\vec{\gamma},\vec{\beta})| \leq \epsilon$, the algorithm terminates. We also set a step tolerance δ, so that the algorithm terminates if the new parameter point is very close to the previous one $|\vec{\gamma}' - \vec{\gamma}|^2 + |\vec{\beta}' - \vec{\beta}|^2 \leq \delta^2$. For gradient-based optimization algorithms such as BFGS, we also use δ as the increment size for estimating gradients via the finite-difference method: $\partial F_p/\partial\gamma_i \simeq [\bar{F}_{p,M'}(\gamma_i + \delta) - \bar{F}_{p,M}(\gamma_i)]/\delta$. In our simulations, we use the BFGS algorithm implemented as fminunc in the standard library of MATLAB R2017b.

Using the approach described above, we simulate experiments of optimizing the QAOA with measurement projection noise for a few example instances, with various choices of precision parameters (ϵ, ξ, δ) and initial points. For the representative instance studied in Fig. 7, we set $\epsilon = 0.1$, $\xi = 0.05$, and $\delta = 0.01$. In each run of the simulated experiment, we start with the QAOA of level either $p = 1$ or $p = 5$ and optimize increasing levels of the QAOA using our FOURIER$[\infty, 0]$ heuristic strategy. The initial point of QAOA optimization is either randomly selected (when starting at $p = 1$) or chosen based on an educated guess using optimal parameters from small-sized instances (at $p = 1$ and $p = 5$). Specifically, the educated guess for the initial points we use are $(\vec{u}^0, \vec{v}^0) = (1.4849, 0.5409)$ at level $p = 1$ and

$$\vec{u}^0 = (1.9212, 0.2891, 0.1601, 0.0564, 0.0292), \tag{F3}$$

$$\vec{v}^0 = (0.6055, -0.0178, 0.0431, -0.0061, 0.0141) \tag{F4}$$

at level $p = 5$. For each such run, we keep track of the history of all the measurements, so that the largest cut Cut_i found after the ith measurement can be calculated. We repeat each experiment for 500 times with different pseudorandom number generation seeds and average over their histories.

APPENDIX G: TECHNIQUES TO SPEED UP NUMERICAL SIMULATION

In this Appendix, we discuss a number of techniques we exploit to speed up the numerical simulation for both the QAOA and QA.

First, we make use of the symmetries present in the Hamiltonian. For general graphs, the only symmetry operator that commutes with both H_C and H_B is the parity operator $\mathcal{P} = \prod_{i=1}^N \sigma_i^x$: $[H_C, \mathcal{P}] = 0, [H_B, \mathcal{P}] = 0$, and so does $[H_{\mathrm{QA}}(s), \mathcal{P}] = 0$, where $H_{\mathrm{QA}}(s)$ is the quantum annealing Hamiltonian in Eq. (9). The parity operator has two eigenvalues, $+1$ and -1, each with half of the entire Hilbert space. The initial state for both the QAOA and QA is in the positive sector, i.e., $\mathcal{P}|+\rangle^{\otimes N} = |+\rangle^{\otimes N}$. Thus, any dynamics remain in the positive parity sector. We can rewrite H_C and H_B in the basis of the eigenvectors of $\mathcal{P}$ and reduce the Hilbert space from 2^N to 2^{N-1} by working in the positive parity sector.

For QA, dynamics involving the time-dependent Hamiltonian can be simulated by dividing the total simulation time T into sufficiently small discrete time τ and implement each time step sequentially. At each small step, one can evolve the state without forming the full evolution operator [72], either using the Krylov subspace projection method [73] or a truncated Taylor series approximation [74]. In our simulation, we use a scaling and squaring method with a truncated Taylor series approximation [74], as it appears to run slightly faster than the Krylov subspace method for small time steps.

For the QAOA, the dynamics can be implemented in a more efficient way due to the special form of the operators H_C and H_B. We work in the standard σ_z basis. Thus, $H_C = \sum_{\langle i,j\rangle}(w_{ij}/2)(\mathbb{1} - \sigma_i^z\sigma_j^z)$ can be written as a diagonal matrix, and the action of $e^{-i\gamma H_C}$ can be implemented as vector operations. For H_B, the time evolution operator can be simplified as

$$e^{-i\beta H_B} = \prod_{j=1}^N e^{-i\beta\sigma_j^x} = \prod_{j=1}^N (\mathbb{1}\cos\beta - i\sigma_j^x \sin\beta). \tag{G1}$$

Therefore, the action of $e^{-i\beta H_B}$ can also be implemented as N sequential vector operations without explicitly forming the sparse matrix H_B, which both improves simulation speed and saves memory. In addition, in the optimization of variational parameters, we calculate the gradient analytically instead of using finite-difference methods. Techniques similar to the gradient ascent pulse engineering method [75] are used, which reduces the cost of computing the gradient from $O(p^2)$ to $O(p)$, for a p-level QAOA. Finally, in our FOURIER strategy, we need to calculate the gradient of the objective function with respect to the new parameters $(\vec{u}, \vec{v})$. Since $\vec{\gamma} = A_S\vec{u}$ and $\vec{\beta} = A_C\vec{v}$ for some matrices A_S and A_C, their gradients are also related via $\nabla_{\vec{u}} = A_S\nabla_{\vec{\gamma}}$ and $\nabla_{\vec{v}} = A_S\nabla_{\vec{\beta}}$.

APPENDIX H: QAOA FOR MAXIMUM INDEPENDENT SET

In this Appendix, we briefly illustrate the generality of our results by applying the QAOA to another class of combinatorial optimization problems called the MIS. Given a graph $G = (V, E)$, the MIS problem concerns finding the largest independent set—a subset of vertices where no two of which share an edge. In other words, the problem Hamiltonian is

$$H_P = \mathcal{P}_{\mathrm{IS}}\left(\sum_i \hat{n}_i\right)\mathcal{P}_{\mathrm{IS}}, \tag{H1}$$

where $\hat{n} = |1\rangle\langle 1| = (\mathbb{1} - \sigma^z)/2$ and $\mathcal{P}_{\mathrm{IS}}$ is the projection (or restriction) onto the subspace of independent set states $\mathrm{span}\{|\psi\rangle : \hat{n}_i\hat{n}_j|\psi\rangle = 0 \ \forall \ (i,j) \in E\}$. The p-level QAOA for the MIS, first suggested by Ref. [2], involves the preparation of the following variational wave function:

$$|\psi_p(\vec{\gamma},\vec{\beta})\rangle = e^{-i\beta_p H_Q}\prod_{k=1}^{p-1} e^{-i\gamma_k H_P}e^{-i\beta_k H_Q}|0\rangle^{\otimes N}, \tag{H2}$$

where

$$H_Q = \mathcal{P}_{\mathrm{IS}}\left(\sum_i \sigma_i^x\right)\mathcal{P}_{\mathrm{IS}}. \tag{H3}$$

Similar to the case of MaxCut, here the QAOA works by repeatedly measuring the system in the computational basis to obtain an estimate of $G_p(\vec{\gamma},\vec{\beta}) = \langle\psi_p(\vec{\gamma},\vec{\beta})|H_P|\psi_p(\vec{\gamma},\vec{\beta})\rangle$ and using a classical computer to search for the best variational parameters $(\vec{\gamma}^*,\vec{\beta}^*)$ so as to maximize G_p. We note that the evolution can be implemented using the following physical Hamiltonian:

$$H^{\mathrm{MIS}}_{\mathrm{physical}}(t) = \sum_i[\Delta(t)\hat{n}_i + \Omega(t)\sigma_i^x] + \sum_{\langle i,j\rangle}U\hat{n}_i\hat{n}_j. \tag{H4}$$

In the $U \gg |\Delta|, |\Omega|$ limit, the system is constrained to the manifold where $\hat{n}_i\hat{n}_j = 0$ for all edges $\langle i,j\rangle$ (since the initial state is $|0\rangle^{\otimes N}$), and the QAOA circuit can be performed by setting appropriate waveforms of $\Delta(t)$ and $\Omega(t)$. See Ref. [46] for an implementation scheme of the MIS with a platform of neutral atoms interacting via Rydberg excitations.

After performing exhaustive search of QAOA parameters using randomly initialized optimization for many instances of the MIS, we discover a similar pattern. This pattern is illustrated in Fig. 15(a), where we see that the optimal parameters at $p = 3$ tend to cluster in two visually distinct groups. For one of the groups, the smooth curve underlying the parameters exhibits a resemblance to a quantum annealing protocol, using a time-dependent Hamiltonian:

$$H^{\mathrm{MIS}}_{\mathrm{QA}}(t) = f_P(t)H_P + f_Q(t)H_Q. \tag{H5}$$

For example, if we choose (f_P, f_Q) such that $f_P(0) > 0$ and $f_P(T) < 0$, and $f_Q(0) = f_Q(T) = 0$, then we can initialize the system in $|0\rangle^{\otimes N}$, which is the ground state of $H^{\mathrm{MIS}}_{\mathrm{QA}}(t = 0)$, and evolve adiabatically to reach the ground state of $H^{\mathrm{MIS}}_{\mathrm{QA}}(t = T) \propto -H_P$ (i.e., the state encoding the MIS solution). The Hamiltonian H_Q, which is

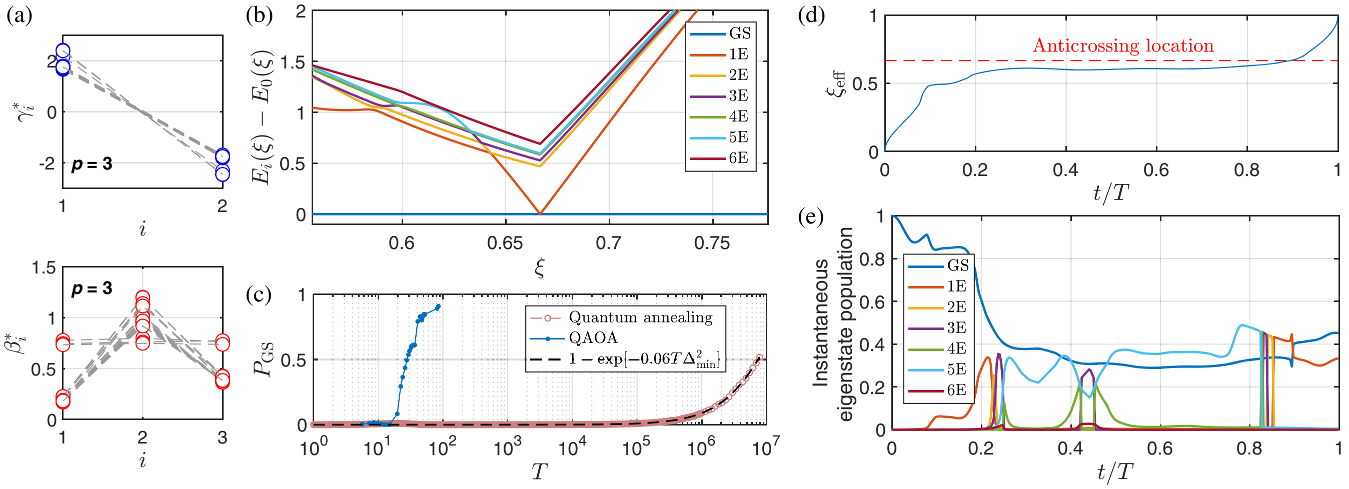

FIG. 15. (a) Pattern in the optimal QAOA parameters at level $p = 3$ for 20 random instances of the MIS problem. Each dashed line connects parameters for one particular graph instance. (b) The energy difference relative to the ground state in the quantum annealing protocol of Eq. (H6) for an example 32-vertex MIS instance. The annealing protocol progresses as the parameter ξ increases from 0 to 1. The minimum spectral gap between the ground state (GS) and the first excited state (1E) is $\Delta_{\min} = 0.0012$ at $\xi = 0.6666$. (c) Comparing performance of quantum annealing and the QAOA on the example instance, in terms of the ground state population at the end of the quantum evolution. The equivalent evolution time for the QAOA is calculated via $T_{\mathrm{QAOA}} = \sum_{i=1}^{p-1}|\gamma_i| + \sum_{i=1}^{p}|\beta_i|$. (d) The effective annealing schedule converted from optimized 25-level QAOA parameters for the example MIS instance. (e) The population of the system in the instantaneous eigenstates, during the effective annealing schedule that approximates the dynamics under the 25-level QAOA. Here, we observe that the algorithm attempts to transport the system to the fifth excited state, keeping it there before it undergoes a series of anticrossings toward the ground state.

turned on in the middle of the time evolution, induces couplings between different independent-set basis states and opens a spectral gap between the ground state and excited states. For concreteness of discussion, we focus on the following choice of annealing schedule:

$$f_P^c(\xi) = 6(1 - 2\xi), \qquad f_Q^c(\xi) = \sin^2(\pi\xi), \qquad \xi(t) = t/T. \tag{H6}$$

We further analyze the performance and mechanism of the QAOA for the MIS by focusing on example instances that are difficult for adiabatic quantum annealing due to small spectral gaps. In Fig. 15(b), we show the level-crossing structure for such an example instance, where the minimum spectral gap is $\Delta_{\min} = 0.0012$. The same instance is studied in Ref. [46]. To study the performance of the QAOA in deeper-depth circuits, we use the interpolation-based heuristic strategy outlined in Appendix B 1 to optimize QAOA parameters for this example instance starting at level $p = 3$. The performance of the QAOA and quantum annealing are then compared in Fig. 15(c), where we see that the QAOA is able to obtain a much larger ground state population in a much shorter time compared to the adiabatic timescale of $1/\Delta_{\min}^2 \approx 10^6$. We then study the mechanism of the QAOA by converting its parameters at level $p = 25$ to a smooth annealing path $(f_P^{\mathrm{QAOA}}, f_Q^{\mathrm{QAOA}})$ in a similar procedure as in Sec. V B. This annealing path is visualized in Fig. 15(d), where we plot the effective $\xi_{\mathrm{eff}}(t)$ defined by $f_P^c[\xi_{\mathrm{eff}}(t)]/f_Q^c[\xi_{\mathrm{eff}}(t)] = f_P^{\mathrm{QAOA}}(t)/f_Q^{\mathrm{QAOA}}(t)$ for the QAOA-like schedule. We then monitor populations in the instantaneous eigenstates during the evolution to gain insights into the mechanism of the QAOA. As shown in Fig. 15(e), the QAOA is able to learn to navigate a very complicated level-crossing structure by a combination of adiabatic and nonadiabatic operations: The system diabatically couples to the excited states, lingers to maximize population in the fifth excited state, and then exploits a series of anticrossings to return to the ground state. Our results here demonstrate that the nonadiabatic mechanisms observed in the QAOA for MaxCut can play a significant role in more general problems, such as difficult MIS instances.

[1] J. Preskill, *Quantum Computing in the NISQ Era and Beyond*, Quantum **2**, 79 (2018).

[2] E. Farhi, J. Goldstone, and S. Gutmann, *A Quantum Approximate Optimization Algorithm*, arXiv:1411.4028.

[3] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'Brien, *A Variational Eigenvalue Solver on a Photonic Quantum Processor*, Nat. Commun. **5**, 4213 (2014).

[4] N. Moll, P. Barkoutsos, L. S. Bishop, J. M. Chow, A. Cross, D. J. Egger, S. Filipp, A. Fuhrer, J. M. Gambetta, M. Ganzhorn, A. Kandala, A. Mezzacapo, P. Müller, W. Riess, G. Salis, J. Smolin, I. Tavernelli, and K. Temme, *Quantum Optimization Using Variational Algorithms on Near-Term Quantum Devices*, Quantum Sci. Technol. **3**, 030503 (2018).

[5] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta, *Hardware-Efficient Variational Quantum Eigensolver for Small Molecules and Quantum Magnets*, Nature (London) **549**, 242 (2017).

[6] J. S. Otterbach *et al.*, *Unsupervised Machine Learning on a Hybrid Quantum Computer*, arXiv:1712.05771.

[7] C. Kokail, C. Maier, R. van Bijnen, T. Brydges, M. K. Joshi, P. Jurcevic, C. A. Muschik, P. Silvi, R. Blatt, C. F. Roos, and P. Zoller, *Self-Verifying Variational Quantum Simulation of the Lattice Schwinger Model*, Nature (London) **569**, 355 (2019).

[8] X. Qiang, X. Zhou, J. Wang, C. M. Wilkes, T. Loke, S. O'Gara, L. Kling, G. D. Marshall, R. Santagati, T. C. Ralph, J. B. Wang, J. L. O'Brien, M. G. Thompson, and J. C. F. Matthews, *Large-Scale Silicon Quantum Photonics Implementing Arbitrary Two-Qubit Processing*, Nat. Photonics **12**, 534 (2018).

[9] G. Pagano, A. Bapat, P. Becker, K. S. Collins, A. De, P. W. Hess, H. B. Kaplan, A. Kyprianidis, W. L. Tan, C. Baldwin, L. T. Brady, A. Deshpande, F. Liu, S. Jordan, A. V. Gorshkov, and C. Monroe, *Quantum Approximate Optimization with a Trapped-Ion Quantum Simulator*, arXiv:1906.02700.

[10] E. Farhi, J. Goldstone, and S. Gutmann, *A Quantum Approximate Optimization Algorithm Applied to a Bounded Occurrence Constraint Problem*, arXiv:1412.6062.

[11] E. Farhi and A. W. Harrow, *Quantum Supremacy through the Quantum Approximate Optimization Algorithm*, arXiv:1602.07674.

[12] M. B. Hastings, *Classical and Quantum Bounded Depth Approximation Algorithms*, arXiv:1905.07047.

[13] S. Bravyi, A. Kliesch, R. Koenig, and E. Tang, *Obstacles to State Preparation and Variational Optimization from Symmetry Protection*, arXiv:1910.08980.

[14] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, *Barren Plateaus in Quantum Neural Network Training Landscapes*, Nat. Commun. **9**, 4812 (2018).

[15] E. Crosson, E. Farhi, C. Y.-Yu. Lin, H.-H. Lin, and P. Shor, *Different Strategies for Optimization Using the Quantum Adiabatic Algorithm*, arXiv:1401.7320.

[16] S. Muthukrishnan, T. Albash, and D. A. Lidar, *Tunneling and Speedup in Quantum Optimization for Permutation-Symmetric Problems*, Phys. Rev. X **6**, 031010 (2016).

[17] L. Hormozi, E. W. Brown, G. Carleo, and M. Troyer, *Nonstoquastic Hamiltonians and Quantum Annealing of an Ising Spin Glass*, Phys. Rev. B **95**, 184416 (2017).

[18] T. Albash and D. A. Lidar, *Adiabatic Quantum Computation*, Rev. Mod. Phys. **90**, 015002 (2018).

[19] H. Bernien, S. Schwartz, A. Keesling, H. Levine, A. Omran, H. Pichler, S. Choi, A. S. Zibrov, M. Endres, M. Greiner, V. Vuletić, and M. D. Lukin, *Probing Many-Body Dynamics on a 51-Atom Quantum Simulator*, Nature (London) **551**, 579 (2017).

[20] M. Saffman, T. G. Walker, and K. Mølmer, *Quantum Information with Rydberg Atoms*, Rev. Mod. Phys. **82**, 2313 (2010).

[21] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity* (Courier, North Chelmsford, 1998).

[22] B. Korte, J. Vygen, B. Korte, and J. Vygen, *Combinatorial Optimization* (Springer, New York, 2012), Vol. 2.

[23] J. Håstad, *Some Optimal Inapproximability Results*, J. ACM **48**, 798 (2001).

[24] P. Berman and M. Karpinski, *On Some Tighter Inapproximability Results (Extended Abstract)* (Springer, Berlin, 1999), pp. 200–209.

[25] M. X. Goemans and D. P. Williamson, *Improved Approximation Algorithms for Maximum Cut and Satisfiability Problems Using Semidefinite Programming*, J. ACM **42**, 1115 (1995).

[26] E. Halperin, D. Livnat, and U. Zwick, *MAX CUT in Cubic Graphs*, J. Algorithms **53**, 169 (2004).

[27] 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).

[28] D. Wecker, M. B. Hastings, and M. Troyer, *Training a Quantum Optimizer*, Phys. Rev. A **94**, 022309 (2016).

[29] 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).

[30] W. W. Ho and T. H. Hsieh, *Efficient Unitary Preparation of Non-trivial Quantum States*, SciPost Phys. **6**, 029 (2019).

[31] 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).

[32] E. R. Anschuetz, J. P. Olson, A. Aspuru-Guzik, and Y. Cao, *Variational Quantum Factoring*, arXiv:1808.08927.

[33] The approximation ratio $r = (2p+1)/(2p+2)$ is found for an infinite ring. For a finite ring with N vertices, numerical calculations show that for $p < N/2$ one has $r = (2p+1)/(2p+2)$ for even N and $r = \{[(2p+1)(N+1)]/[(2p+2)N]\}$ for odd N, and for $p \geq N/2$ one has $r = 1$.

[34] The initial points β_i^0 are drawn uniformly from $[-(\pi/4), (\pi/4))$, and γ_i^0 are drawn uniformly $[-(\pi/2), (\pi/2))$ for u3R graphs or $[-2\pi, 2\pi)$ for w3R graphs. Although γ_i^0 can meaningfully take values beyond the restricted range $\gamma_i^0 \in [-2\pi, 2\pi)$ for a w3R graph, we find that broadening the range does not improve the performance. The ranges of the output parameters are not restricted in our unconstrained optimization routine.

[35] C. G. Broyden, *The Convergence of a Class of Double-Rank Minimization Algorithms 1. General Considerations*, IMA J. Appl. Math. **6**, 76 (1970); R. Fletcher, *A New Approach to Variable Metric Algorithms*, Comput. J. **13**, 317 (1970); D. Goldfarb, *A Family of Variable-Metric Methods Derived by Variational Means*, Math. Comput. **24**, 23 (1970); D. F. Shanno, *Conditioning of Quasi-Newton Methods for Function Minimization*, Math. Comput. **24**, 647 (1970).

[36] We denote the best of all local optima optimized from k random initial points (seeds) to be $(\vec{\gamma}^{B[k]}, \vec{\beta}^{B[k]})$. When the same optimum $(\vec{\gamma}^{B[k]}, \vec{\beta}^{B[k]})$ is found from many different seeds and continues to yield the best $F_p(\vec{\gamma}^{B[k]}, \vec{\beta}^{B[k]})$ as we increase the number of seeds k, we then claim that it is a global optimum, i.e., $(\vec{\gamma}^*, \vec{\beta}^*) = \lim_{k\to\infty}(\vec{\gamma}^{B[k]}, \vec{\beta}^{B[k]})$.

[37] J. A. Nelder and R. Mead, *A Simplex Method for Function Minimization*, Comput. J. **7**, 308 (1965).

[38] P. I. Frazier, *A Tutorial on Bayesian Optimization*, arXiv:1807.02811.

[39] L. D. Landau, *Zur Theorie der Energieubertragung II*, Z. Sowjetunion **2**, 46 (1932); C. Zener, *Non-Adiabatic Crossing of Energy Levels*, Proc. R. Soc. A **137**, 696 (1932).

[40] T. Kadowaki and H. Nishimori, *Quantum Annealing in the Transverse Ising Model*, Phys. Rev. E **58**, 5355 (1998).

[41] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, and D. Preda, *A Quantum Adiabatic Evolution Algorithm Applied to Random Instances of an NP-Complete Problem*, Science **292**, 472 (2001).

[42] S. Boixo, T. F. Rønnow, S. V. Isakov, Z. Wang, D. Wecker, D. A. Lidar, J. M. Martinis, and M. Troyer, *Evidence for Quantum Annealing with More than One Hundred Qubits*, Nat. Phys. **10**, 218 (2014).

[43] T. F. Rønnow, Z. Wang, J. Job, S. Boixo, S. V. Isakov, D. Wecker, J. M. Martinis, D. A. Lidar, and M. Troyer, *Defining and Detecting Quantum Speedup*, Science **345**, 420 (2014).

[44] To be consistent with the language of QA, here we use the terminology of ground state and low excited states of $-H_C$ instead of referring to them as highest excited states in the MaxCut language.

[45] This result could change with different normalizations of the Hamiltonians H_C and H_B, but our qualitative results remain the same. The physical limitation in the experiment is typically the interaction strength in H_C.

[46] H. Pichler, S.-T. Wang, L. Zhou, S. Choi, and M. D. Lukin, *Quantum Optimization for Maximum Independent Set Using Rydberg Atom Arrays*, arXiv:1808.10816.

[47] H. Pichler, S.-T. Wang, L. Zhou, S. Choi, and M. D. Lukin, *Computational Complexity of the Rydberg Blockade in Two Dimensions*, arXiv:1809.04954.

[48] G. G. Guerreschi and M. Smelyanskiy, *Practical Optimization for Hybrid Quantum-Classical Algorithms*, arXiv:1701.01450.

[49] F. Rendl, G. Rinaldi, and A. Wiegele, *Solving Max-Cut to Optimality by Intersecting Semidefinite and Polyhedral Relaxations*, Math. Program. **121**, 307 (2010).

[50] U. Benlic and J.-K. Hao, *Breakout Local Search for the Max-Cut Problem*, Engineering Applications of Artificial Intelligence **26**, 1162 (2013).

[51] E. Cuthill and J. McKee, *Reducing the Bandwidth of Sparse Symmetric Matrices*, in *Proceedings of the 24th National Conference, ACM '69* (Association for Computing Machinery, New York, 1969), pp. 157–172.

[52] L. Lin and Y. Lin, *Square-Root Rule of Two-Dimensional Bandwidth Problem*, RAIRO, Theor. Inf. Appl. **45**, 399 (2011).

[53] W. Lechner, P. Hauke, and P. Zoller, *A Quantum Annealing Architecture with All-to-All Connectivity from Local Interactions*, Sci. Adv. **1**, e1500838 (2015).

[54] J. Díaz and M. Kamiński, *Max-Cut and Max-Bisection Are NP-Hard on Unit Disk Graphs*, Theor. Comput. Sci. **377**, 271 (2007).

[55] H. Labuhn, D. Barredo, S. Ravets, S. de Léséleuc, T. Macrì, T. Lahaye, and A. Browaeys, *Tunable Two-Dimensional*

Arrays of Single Rydberg Atoms for Realizing Quantum Ising Models, Nature (London) **534**, 667 (2016).

[56] A. Keesling, A. Omran, H. Levine, H. Bernien, H. Pichler, S. Choi, R. Samajdar, S. Schwartz, P. Silvi, S. Sachdev, P. Zoller, M. Endres, M. Greiner, V. Vuletic, and M. D. Lukin, *Probing Quantum Critical Dynamics on a Programmable Rydberg Simulator*, Nature (London) **568**, 207 (2019).

[57] A. Kumar, T.-Y. Wu, F. Giraldo, and D. S. Weiss, *Sorting Ultracold Atoms in a Three-Dimensional Optical Lattice in a Realization of Maxwell's Demon*, Nature (London) **561**, 83 (2018).

[58] H. Häffner, C. F. Roos, and R. Blatt, *Quantum Computing with Trapped Ions*, Phys. Rep. **469**, 155 (2008).

[59] S. Debnath, N. M. Linke, C. Figgatt, K. A. Landsman, K. Wright, and C. Monroe, *Demonstration of a Small Programmable Quantum Computer with Atomic Qubits*, Nature (London) **536**, 63 (2016).

[60] J. Zhang, G. Pagano, P. W. Hess, A. Kyprianidis, P. Becker, H. Kaplan, A. V. Gorshkov, Z. X. Gong, and C. Monroe, *Observation of a Many-Body Dynamical Phase Transition with a 53-Qubit Quantum Simulator*, Nature (London) **551**, 601 (2017).

[61] R. Barends *et al.*, *Digitized Adiabatic Quantum Computing with a Superconducting Circuit*, Nature (London) **534**, 222 (2016).

[62] C. Neill *et al.*, *A Blueprint for Demonstrating Quantum Supremacy with Superconducting Qubits*, Science **360**, 195 (2018).

[63] H. Levine, A. Keesling, A. Omran, H. Bernien, S. Schwartz, A. S. Zibrov, M. Endres, M. Greiner, V. Vuletić, and M. D. Lukin, *High-Fidelity Control and Entanglement of Rydberg-Atom Qubits*, Phys. Rev. Lett. **121**, 123603 (2018).

[64] L. Isenhower, M. Saffman, and K. Mølmer, *Multibit CNOT Quantum Gates via Rydberg Blockade*, Quantum Inf. Process. **10**, 755 (2011).

[65] T. Inagaki, Y. Haribara, K. Igarashi, T. Sonobe, S. Tamate, T. Honjo, A. Marandi, P. L. McMahon, T. Umeki, K. Enbutsu, O. Tadanaga, H. Takenouchi, K. Aihara, K.-i. Kawarabayashi, K. Inoue, S. Utsunomiya, and H. Takesue, *A Coherent Ising Machine for 2000-Node Optimization Problems*, Science **354**, 603 (2016).

[66] P. L. McMahon, A. Marandi, Y. Haribara, R. Hamerly, C. Langrock, S. Tamate, T. Inagaki, H. Takesue, S. Utsunomiya, K. Aihara, R. L. Byer, M. M. Fejer, H. Mabuchi, and Y. Yamamoto, *A Fully-Programmable 100-Spin Coherent Ising Machine with All-to-All Connections*, Science **354**, 614 (2016).

[67] R. Hamerly *et al.*, *Experimental Investigation of Performance Differences between Coherent Ising Machines and a Quantum Annealer*, Sci. Adv. **5**, eaau0823 (2019).

[68] M. Bukov, A. G. R. Day, D. Sels, P. Weinberg, A. Polkovnikov, and P. Mehta, *Reinforcement Learning in Different Phases of Quantum Control*, Phys. Rev. X **8**, 031086 (2018).

[69] A. Vahdatpour, F. Dabiri, M. Moazeni, and M. Sarrafzadeh, *Theoretical Bound and Practical Analysis of Connected Dominating Set in Ad Hoc and Sensor Networks*, in *Distributed Computing* (Springer, Berlin, 2008), pp. 481–495.

[70] P. K. Agarwal, M. van Kreveld, and S. Suri, *Label Placement by Maximum Independent Set in Rectangles*, Comput. Geom. **11**, 209 (1998).

[71] G. E. Crooks, *Performance of the Quantum Approximate Optimization Algorithm on the Maximum Cut Problem*, arXiv:1811.08419.

[72] C. Moler and C. Van Loan, *Nineteen Dubious Ways to Compute the Exponential of a Matrix*, Twenty-Five Years Later, SIAM Rev. **45**, 3 (2003).

[73] R. B. Sidje, *Expokit: A Software Package for Computing Matrix Exponentials*, ACM Trans. Math. Softw. **24**, 130 (1998).

[74] A. H. Al-Mohy and N. J. Higham, *Computing the Action of the Matrix Exponential, with an Application to Exponential Integrators*, SIAM J. Sci. Comput. **33**, 488 (2011).

[75] N. Khaneja, T. Reiss, C. Kehlet, T. Schulte-Herbrüggen, and S. J. Glaser, *Optimal Control of Coupled Spin Dynamics: Design of NMR Pulse Sequences by Gradient Ascent Algorithms*, J. Magn. Reson. **172**, 296 (2005).