Parameter Transfer for Quantum Approximate Optimization of Weighted MaxCut

RUSLAN SHAYDULIN, Argonne National Laboratory and JPMorgan Chase
PHILLIP C. LOTSHAW, Oak Ridge National Laboratory
JEFFREY LARSON, Argonne National Laboratory
JAMES OSTROWSKI, University of Tennessee at Knoxville
TRAVIS S. HUMBLE, Oak Ridge National Laboratory

Finding high-quality parameters is a central obstacle to using the quantum approximate optimization algorithm (QAOA). Previous work partially addresses this issue for QAOA on unweighted MaxCut problems by leveraging similarities in the objective landscape among different problem instances. However, we show that the more general weighted MaxCut problem has significantly modified objective landscapes, with a proliferation of poor local optima. Our main contribution is a simple rescaling scheme that overcomes these deleterious effects of weights. We show that for a given QAOA depth, a single “typical” vector of QAOA parameters can be successfully transferred to weighted MaxCut instances. This transfer leads to a median decrease in the approximation ratio of only 2.0 percentage points relative to a considerably more expensive direct optimization on a dataset of 34,701 instances with up to 20 nodes and multiple weight distributions. This decrease can be reduced to 1.2 percentage points at the cost of only 10 additional QAOA circuit evaluations with parameters sampled from a pretrained metadistribution, or the transferred parameters can be used as a starting point for a single local optimization run to obtain approximation ratios equivalent to those achieved by exhaustive optimization in 96.35% of our cases.

CCS Concepts: • Hardware → Quantum Computation; • Computer systems organization → Quantum computing;

Additional Key Words and Phrases: QAOA, parameter optimization, weighted MaxCut

R. Shaydulin and P. C. Lotshaw contributed equally to this research.

P. C. Lotshaw, J. Ostrowski, and T. S. Humble were supported by the Defense Advanced Research Project Agency ONISQ program under award W911NF-20-2-0051. J. Ostrowski was supported by Air Force Office of Scientific Research award AF-FA9550-19-1-0147 and National Science Foundation award OMA-1937008. R. Shaydulin and J. Larson were supported by the U.S. Department of Energy (DOE), Office of Science, Office of Advanced Scientific Computing Research AIDE-QC and FAR-QC projects. R. Shaydulin was supported by the Argonne LDRD program under contract number DE-AC02-06CH11357. Computing resources were provided on Bebop, a high-performance computing cluster operated by the Laboratory Computing Resource Center at Argonne National Laboratory.

Authors’ addresses: R. Shaydulin, Argonne National Laboratory, 9700 S. Cass Avenue, Lemont, IL 60439, and JPMorgan Chase, 237 Park Ave, New York, NY 10017; email: ruslan.shaydulin@jpmchase.com; P. C. Lotshaw and T. S. Humble, Oak Ridge National Laboratory, Oak Ridge, TN, USA, 37830; emails: {lotshawpc, humblets}@ornl.gov; J. Larson, Argonne National Laboratory, Lemont, IL 60516; email: jmlarson@anl.gov; J. Ostrowski, University of Tennessee at Knoxville, Knoxville, TN 37996; email: jostrows@utk.edu.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

© 2023 Copyright held by the owner/author(s).
2643-6817/2023/04-ART19 $15.00
https://doi.org/10.1145/3584706

ACM Transactions on Quantum Computing, Vol. 4, No. 3, Article 19. Publication date: April 2023.
1 INTRODUCTION

With the capabilities of quantum computers rapidly improving [1, 2], researchers widely believe that quantum algorithms may soon become competitive with classical state-of-the-art methods on scientifically relevant problems [3]. A promising application domain for near-term quantum computers is combinatorial optimization, with the Quantum Approximate Optimization Algorithm (QAOA) [4, 5] as a leading candidate algorithm. QAOA is a hybrid quantum-classical algorithm that combines a parameterized quantum evolution with a classical method for obtaining high-quality parameters so that the measurement outcomes of the parameterized quantum state correspond to good solutions for the target combinatorial optimization problem.

The performance of QAOA depends crucially on the choice of these parameters. The problem of identifying good QAOA parameters has attracted considerable interest, with large bodies of theoretical [6–9] and computational [10–17] work. Several recent results show that as a function of the parameters, the QAOA energy landscape is nearly instance independent for typical problem instances that come from a reasonable distribution. This makes it possible to transfer parameters from a set of preoptimized problem instances to new instances, thereby removing or reducing the cost of parameter optimization [18–21]. However, these prior results address only unweighted MaxCut problems and the Sherrington–Kirkpatrick (SK) model, which are of limited practical interest.

In this work, we demonstrate parameter transfer for QAOA applied to the weighted MaxCut problem, which has many relevant practical problems as special cases [22], including network community detection [23]. We develop intuition by first examining the impact of weights on the QAOA parameter landscapes and then propose a simple transfer scheme based on this intuition. Our scheme, illustrated in Figure 1, uses a simple rescaling technique to transfer typical (median) optimized parameters from unweighted MaxCut instances to weighted instances with varying sizes and with weights coming from varying distributions. Despite its simplicity, our approach leads to approximation ratios that are competitive with those found by direct (computationally expensive) parameter optimization. We show how the performance can be further improved at the cost of only a few extra QAOA circuit evaluations by sampling from a metadistribution of good parameters learned using Kernel Density Estimation (KDE). If the transferred parameters are used as the starting point for the BFGS optimization algorithm, a single BFGS run recovers approximation ratios equal to (up to high precision) that of QAOA with directly optimized parameters in 96.35% of the cases considered. Our results provide empirical evidence that instance independence of optimal parameters, previously shown for unweighted MaxCut, extends to the weighted case. Our numerical results are for QAOA with one, two, and three layers, although it is likely that the same qualitative behavior will be observed for a higher number of layers. To show the limitations of parameter transfer, we explore a set of graphs with edge weights drawn from a pathological distribution and identify the mechanisms behind the relatively poor performance of the transferred parameters for these instances. Although these cases challenge our direct parameter transfer approach, we find that our parameters can still serve as a good initial point for BFGS optimization even in such instances.

All data in this article is available in the open source software package QAOAKit [24], and we provide code examples of how our findings can be leveraged in practice [25]. Our findings
Parameter Transfer for Quantum Approximate Optimization of Weighted MaxCut

Fig. 1. Example of parameter transfer for QAOA on weighted MaxCut. (a) We use a previous set of $N = 261,080$ optimized, unweighted, 9-node MaxCut instances to compute median ("typical") scaled parameters $(\beta_{\text{median}}, \gamma_{\text{median}})$ for each QAOA depth $p$ (Table 1 and Equation (8)). We develop an approach in Equations (9) and (10) to rescale the $\beta_{\text{median}}$ and $\gamma_{\text{median}}$ to obtain high-quality parameters for new weighted instances such as the 20-node weighted instance in (b). The blue dashed line in (a) indicates directly optimized $\gamma_1$ and in (b) the transferred $\gamma_1$.

Demonstrate that a simple transfer scheme can dramatically reduce the cost of parameter optimization for QAOA in the general weighted case, bringing QAOA one step closer to being able to address problems of practical interest and demonstrate quantum computational advantage.

2 METHODS

2.1 Setup

We begin by briefly defining the relevant concepts and establishing notation. We consider optimization problems where the goal is to find a binary string $z \in \{0, 1\}^n$ that maximizes some objective function $C(z)$. The objective function can be encoded as an $n$-qubit Hamiltonian $C$ that acts on computational basis states as $C|z\rangle = C(z)|z\rangle$. For a given problem instance, we denote the maximum value of the objective function $C_{\text{max}} = \max_z C(z)$ and the minimum $C_{\text{min}} = \min_z C(z)$. In the following numerical experiments, the values $C_{\text{max}}$ and $C_{\text{min}}$ are obtained by complete enumeration.

The quantum approximation optimization algorithm approximately solves such combinatorial optimization problems by preparing a parameterized quantum state

$$|eta, \gamma, C\rangle = \prod_{l=1}^p e^{-i\beta_l B} e^{-i\gamma_l C} |+\rangle^\otimes n,$$

where $B = \sum_{j=1}^n x_j$ is a sum of single-qubit Pauli $x$ operators. The number $p$ is commonly referred to as the QAOA depth. The parameters $\beta = (\beta_1, \ldots, \beta_p)$ and $\gamma = (\gamma_1, \ldots, \gamma_p)$ are chosen so that the (approximate) solution to the optimization problem is retrieved upon measuring $|\beta, \gamma\rangle$. This is typically done by choosing $(\beta, \gamma)$ that maximize the expected objective value of the measurement outcomes:

$$\langle C(\beta, \gamma) \rangle = \langle \beta, \gamma, C | \beta, \gamma, C \rangle = \sum_{z \in \{0, 1\}^n} \Pr(z) C(z).$$

In the following, we follow the work of Brandão [21] and refer to the value in (2) as the “QAOA objective” or simply “objective” for brevity. For a fixed function $C$ and depth $p$, we can vary $\beta$ and $\gamma$ in (2) to produce an objective landscape [21]. We quantify the performance on a given instance for a given $(\beta, \gamma)$ using the approximation ratio

$$r = \frac{\langle C(\beta, \gamma) \rangle - C_{\text{min}}}{C_{\text{max}} - C_{\text{min}}}$$

with $0 \leq r \leq 1$. When discussing the difference between two approximation ratios, we sometimes present the difference in terms of percentage points (p.p.) for clarity of presentation. We say that the gap between $r_1 > r_2$ is $k$ p.p. if $(r_1 - r_2) \times 100 = k$. 

ACM Transactions on Quantum Computing, Vol. 4, No. 3, Article 19. Publication date: April 2023.
In this article, we consider the weighted MaxCut problem, which has many interesting practical problems as special cases [22, 23]. In weighted MaxCut, given an undirected graph $G = (V, E)$, with each edge $(i, j) \in E$ having a weight $w_{i,j}$, the goal is to partition the set of vertices $V$ into two parts to maximize the sum of weights of edges that span both parts. The objective function of MaxCut is represented by the Hamiltonian

$$C_{\text{MaxCut}} = \frac{1}{2} \sum_{(i,j) \in E} w_{i,j} (1 - z_i z_j), \tag{4}$$

where in the unweighted case, $w_{i,j} = 1, \forall (i,j) \in E$. For a given graph, we denote the average absolute edge weight by

$$|w| = \frac{1}{|E|} \sum_{(i,j) \in E} |w_{i,j}|. \tag{5}$$

### 2.2 Objective Landscapes with Weights

We now develop the intuition for how the introduction of weights affects QAOA objective landscapes as a function of the QAOA parameters $(\beta, \gamma)$. We begin by considering an example of a triangle-free graph with depth $p = 1$, where an analytical formula for $\langle C(\beta, \gamma) \rangle$ is available. We then extend to more general instances and show how it motivates our parameter transfer scheme.

We evaluate how the proposed parameter transfer scheme generalizes to the dataset of 34,701 graph instances numerically in Section 3.

#### 2.2.1 Triangle-Free Graph

We begin by considering the simple case of a triangle-free weighted graph with $p = 1$, for which the QAOA objective can be expressed as

$$\langle C(\beta_1, \gamma_1) \rangle = \frac{W}{2} + \frac{\sin 4\beta_1}{4} \sum_{(i,j) \in E} w_{i,j} \sin(w_{i,j} \gamma_1) \left( \prod_{l \in N_i \setminus \{j\}} \cos(w_{l,j} \gamma_1) + \prod_{k \in N_j \setminus \{i\}} \cos(w_{j,k} \gamma_1) \right), \tag{6}$$

where $W = \sum_{(i,j) \in E} w_{i,j}$ and $N_i \setminus \{j\}$ is the neighborhood of vertex $i$ excluding vertex $j$ [26]. With the fixed optimal $\beta_1 = \frac{\pi}{8}$, the objective is composed of terms that oscillate in $\gamma_1$ at frequencies set by the edge weights and may not be periodic for irrational $w_{i,j}$. Figure 2 shows an example of the behavior for a four-node cycle graph, with the individual components $w_{i,j} \sin(w_{i,j} \gamma_1) \cos(w_{j,k} \gamma_1) \sin(4\beta_1)/4$ (Figure 2(a)) and the approximation ratio both given as a function of $\gamma_1$ (Figure 2(b)).

The green dashed-dotted line in Figure 2(b) shows that a high approximation ratio is obtained at $\gamma_1 \approx 0.23\pi$, because of an approximate alignment and maximization of the terms in Figure 2(a).
The appearance of such a maximum at small values of $\gamma_1$ is a generic feature of (6). It arises from an approximate alignment in the phases of the various terms, where small positive values of $\gamma_1$ produce positive values of both $w_{i,j} \sin(w_{i,j} \gamma_1)$ and $\cos(w_{i,j} \gamma_1)$, independent of the sign or magnitude of $w_{i,j}$. By contrast, larger choices of $\gamma_1$ typically give much smaller $r$, since the phases can have varying or canceling oscillation frequencies that depend on the set of edge weights. Overall, this phase alignment for small values of $\gamma_1$ results in small values of $\gamma_1$ typically being a good parameter choice.

Of course, small positive values of $\gamma_1$ do not necessarily maximize (6) globally. We have constructed such an example in Figure 2(b), where the blue dashed-double-dotted line at $\gamma_1 \approx 11.25 \pi$ gives a superior approximation ratio to the best optimum for small values of $\gamma_1$ at $\gamma_1 \approx 0.23 \pi$. The peak at $\gamma \approx 11.25 \pi$ is obtained by simultaneously approximately maximizing terms with the two prefactors $w_{i,j}$ of largest magnitude in Figure 2(a), which gives a higher objective value than does the approximate alignment of terms at a small value of $\gamma_1$. Similar maxima at large $\gamma_1$ can be devised for other triangle-free graphs with $p = 1$ using (6). In general, however, finding such maxima requires optimizing over an unbounded domain. Therefore, in numerical experiments, we restrict the starting points for the local optimizations to $-\pi/|w| \leq \gamma_1 \leq \pi/|w|$.

2.2.2 Generic Graph. We now extend the motivation from the triangle-free case to generic graphs. In the triangle-free $p = 1$ case, we found a maximum in $r$ at small values of $\gamma_1$ that we attributed to the approximate phase alignment in the component terms of the QAOA objective in (6). In more general cases, the objective is instead given by (2). This equation depends on the operators in (1), which at the $l$th step can be expressed in terms of trigonometric functions of the parameters and weights:

$$e^{-i\beta_l B} = \prod_{j=1}^{n} \left[ \cos \beta_l - ix_j \sin \beta_l \right], \quad \text{and} \quad e^{-i\gamma_l C} = \prod_{(i,j) \in E} \left[ \cos \frac{y_l w_{i,j}}{2} + iz_j z_j \sin \frac{y_l w_{i,j}}{2} \right]. \quad (7)$$

As in (6), the objective value is again determined by terms that oscillate in $w_{i,j} \gamma_1$. Intuitively, we can expect the phases in these terms to approximately align for small $\gamma_1$, whereas larger $\gamma_1$ may effectively randomize the phases and the overall behavior. Hence, we expect a high-quality maximum from the alignment of terms near $\gamma_1 = 0$. We observe this maximum numerically for all $l = 1, \ldots, p$.

Figure 3 demonstrates how these effects impact the QAOA objective landscape for a single, randomly selected, 14-node (not triangle-free) graph with edge weights drawn from different distributions. We plot the approximation ratio $r$ for $p = 1$ QAOA as a function of $(\beta_1, \gamma_1)$. Figure 3(a) considers an unweighted graph, which has a simple periodic structure, whereas Figure 3(b) shows the same graph but with weights drawn from an exponential distribution with a mean weight of 5. Here we see the appearance of a sharp local maximum at small values of $\gamma_1$ while larger $\gamma_1$ give suboptimal $r$ with small fluctuations. Thus, the high-quality maximum at small $\gamma_1$ from approximate phase alignment is also manifest in this more general example.

The maximum in Figure 3(b) is surrounded by a small basin of attraction, making it difficult to find optimal parameters when starting from random initial parameters. Furthermore, the optimized $\gamma_1$ in Figure 3(b) is much smaller than in Figure 3(a); hence, naively using optimized parameters from Figure 3(a) in Figure 3(b) will result in a poor approximation ratio. To allow the transfer from unweighted to weighted graphs, we must take into account the magnitude of edge weights. Specifically, we observe that an optimal value of $\gamma$ changes with the mean absolute value of the edge weights as $\gamma \approx O(1/|w|)$, irrespective of the depth $p$.

To see this, first consider QAOA applied to two general (not necessarily MaxCut) objective functions $C_w$ and $C$ that are equivalent up to some factor $w > 0$—in other words, $C_w = wC$. If QAOA achieves an approximation ratio of $r$ for $C$ with parameters $(\beta, \gamma)$, then QAOA will achieve an
Fig. 3. Approximation ratio landscapes \( C(\beta_1, \gamma_1) \) at \( p = 1 \) and scans \( C(\beta_1^*, \gamma_1) \) at optimized \( \beta_1^* \) for unweighted MaxCut on a 14-vertex graph (a), weighted MaxCut for the same graph with edge weights drawn from an exponential distribution with mean absolute weight \( |w| = 5 \) (b), exponential (same as (b)) rescaled to have \( |w| = 1 \) (c), and Cauchy distributions (d). The \( \beta_1 \) ranges are the same in all panels.

Identical approximation ratio for \( C_w \) with \( (\beta, \gamma/w) \). We give a rigorous proof of this statement in Appendix A. Note that this statement holds for arbitrary depth \( p \).

This observation informs our intuition about typical problems and cases. If the objective function is on average increased by a factor \( w > 0 \), then \( \gamma^* \) should decrease by the same factor. Equivalently, the objective function can be rescaled and \( \gamma^* \) fixed. In the case of MaxCut, the objective function scales linearly with graph edge weights; therefore, the relevant metric is the average absolute edge weight \( |w| \).

In Figure 3(c), we demonstrate how weight magnitudes affect the objective landscape. The figure shows the same weight instance as Figure 3(b) but with a uniform rescaling in \( C \) such that \( |w| = 1 \) (the same plot could also be generated by fixing \( C \) and rescaling \( \gamma_1 \)). This "stretches" the landscape, alleviating the issue of a narrow basin of attraction in Figure 3(b). The objective has a similar structure to the unweighted case near \( \gamma_1 = 0 \), which has the same average edge weight. The optimized parameters from the unweighted case can now be transferred to the weighted case. We find similar behavior to Figure 3(b) and (c) both for weights drawn from a uniform distribution of positive values and for weights drawn from a uniform distribution of positive and negative values. The optimized \( \gamma_1 \) from the unweighted case can be successfully transferred to each of these varying weighted cases. The scaling factor \( |w| \) is essential in achieving the correspondence between landscapes that enables parameter transfer.

The final plot in Figure 3(d) shows an instance with weights drawn from a Cauchy distribution. The Cauchy distribution on \( [-\infty, \infty] \) is well known as an exemplar of pathological behavior, with symmetry about the origin and normalization but with an undefined mean and higher moments. In this work, we use a truncated Cauchy distribution, with weights \( -10^3 \leq w_{i,j} \leq 10^3 \) before rescaling to \( |w| = 1 \), which avoids the pathologies of the moments but achieves a much higher variance for the edge weights than other distributions considered so far. The landscape is visibly "rugged" with many local maxima, although we still observe a prominent maximum near \( \gamma_1 = 0 \). In the following, we show evidence that even for these instances we can often determine high-quality parameters by transferring the optimized parameters from the unweighted case and using them as a starting point for a local optimizer.

To summarize, we identified two important aspects of the objective landscapes for weighted graphs. The first is the appearance of a prominent local maximum at small \( \gamma_1 \) values, related to phase alignment in components of the objective. The second is the overall magnitude of the objective function, which can be standardized between instances by rescaling all edge weights by a
factor $|w|$. When the variance of the edge weights is not too large, this rescaling aligns the objective maximum of weighted instances at small $\gamma$ with the corresponding maximum of unweighted instances, allowing parameter transfer. The examples we give are for $p = 1$. We expect similar arguments and intuition to apply to $p > 1$, and we present numerical evidence for $p \in \{1, 2, 3\}$ in Section 3.

### 2.3 Scaling Rule for Parameter Transfer

We now present our transfer procedure. An overview is shown in Figure 1. As the intuition developed earlier suggests, to transfer QAOA parameters from unweighted to weighted MaxCut instances, we must rescale the parameters to account for the change in the objective function values. We account for this change by incorporating the average edge weight (following the intuition outlined previously) and the average degree of the graph. The parameter scaling with the average degree improves performance when transferring parameters between different graphs; it has been observed previously for the unweighted MaxCut and the SK model [8, 27, 28] and proven rigorously in the infinite-size limit [7, 29].

Consider a dataset of preoptimized parameters $(\beta_j^*, \gamma_j^*)_{j=1}^N$ for $N$ unweighted graphs. We begin by rescaling the parameters in the dataset using the average degree $d_j = \frac{\sum_{k \in V} d_k^j}{|V|}$ of the graph:

$$
(\beta_j^S, \gamma_j^S) = \left( \beta_j^*, \frac{\gamma_j^*}{\arctan\left(\frac{1}{\sqrt{d_j^*}}\right)} \right) \quad \text{for } j = 1, \ldots, N.
$$

We scale $\gamma$ by $\arctan\left(\frac{1}{\sqrt{d_j^*}}\right)$ following the exact formula for optimal triangle-free and $p = 1$ $\gamma$ from Wang et al. [28] because we observe it to empirically give a better performance for the low values of $p$ considered in this work. The use of the arctan scaling here is a heuristic choice, as we consider non-triangle-free and $p > 1$ instances for transfer, which do not conform to the analytic assumptions of Wang et al. [28]. Nonetheless, we find this same scaling to be useful in achieving high-quality results for our cases. For higher values of $p$, scaling by $\sqrt{d}$ may be preferable [7, 27, 29], although the two are asymptotically equivalent for large $d$. Next, the median scaled parameters $(\beta^S_{\text{median}}, \gamma^S_{\text{median}})$ are computed from $(\beta_j^S, \gamma_j^S)_{j=1}^N$. This completes steps pictured in Figure 1(a).

Finally, to determine transferred angles for a new instance, the precomputed median parameters are rescaled as

$$
\beta_w = \beta^S_{\text{median}},
$$

$$
\gamma_w = \gamma^S_{\text{median}} \frac{\arctan\left(\frac{1}{\sqrt{d_w}}\right)}{|w|}.
$$

Here, $d_w$ is the average degree and $|w|$ is the average absolute edge weight of the weighted graph to which the parameters are transferred. Equations (9) and (10) determine the transferred parameters as in Figure 1(b).

In this work, we use the dataset containing preoptimized parameters for all $N = 261,080$ non-isomorphic connected nine-node graphs [18, 24]. The median scaled parameters from this dataset $\beta^S_{\text{median}}$ and $\gamma^S_{\text{median}}$ are given in Table 1. With the precomputed median parameters, it is straightforward to generate transferred angles for new weighted instances using Equations (9) and (10). We evaluate the success of this simple procedure for a variety of new instances in the next section.
3 RESULTS OF NUMERICAL EXPERIMENTS

We evaluate the proposed parameter transfer scheme on a dataset of 34,701 weighted graphs with up to 20 nodes. The dataset includes all nonisomorphic graphs on 8 nodes, 300 Erdős-Rényi random graphs on 14 nodes, and 50 Erdős-Rényi with 20 nodes, where these random graphs have an edge-creation probability of 0.5. The edge weights are drawn from three distributions: uniform on \((0, b)\), uniform on \((-b, b)\), and exponential. Without loss of generality, we set \(b = 1\) to generate weights and then rescale the edge weights in each instance so that \(|w| = 1\). To provide a baseline against which to compare our transfer proposal, we obtain optimized parameters and QAOA approximation ratios by exhaustive optimization for each weighted graph. The precise definition of the weight distributions and an overview of the parameter optimization are given in Appendix A. We release the full dataset in QAOAKit [30, 31].

The median parameters to be transferred are provided in Table 1. This one fixed set of median parameters is then transferred to each weighted graph following Equations (9) and (10). Additionally, we have experimented with using rescaled fixed-angle conjecture parameters from Wurtz and Lykov [20]. Because the median parameters provided better performance, we do not include the results using these fixed-angle conjecture parameters.

The approximation ratios obtained by using transferred and directly numerically optimized parameters are presented in Figure 4. We observe a median (over all \(p\) and weight distributions) difference in approximation ratios between QAOA with directly optimized and transferred median parameters of only 2.0 p.p. As expected, the difference grows with depth \(p\) (1.2 p.p. for \(p = 1\), 2.4 p.p. for \(p = 2\), and 3.0 p.p. for \(p = 3\)). We also observe that the effectiveness of the parameter transfer depends significantly on the distribution of weights: uniform on \((0, 1)\) is the easiest for parameter transfer (1.0 p.p.), uniform on \((-1, 1)\) being harder (2.1 p.p.), and exponentially drawn weights leading to the hardest instances (3.6 p.p.). We observe that transfer works well (the gap is small) if the variance of the weights is low. As the variance increases, the performance of the transfer procedure deteriorates (i.e., the gap increases). The relationship between the performance of QAOA with transferred parameters and the variance of the weights is shown in Figure 5 for \(p = 1\) and \(|V| = 14\) nodes, with a similar behavior observed for other values of \(p\) and \(|V|\). We note that even in the worst case of Figure 4 (\(p = 3\), exponentially drawn weights), the decrease in approximation ratio from using median parameters is only 4.5 p.p. This demonstrates that the concentration of optimized parameters, previously observed for unweighted MaxCut and the SK model, is present in more general weighted MaxCut instances if they are rescaled appropriately.
Fig. 4. Approximation ratios using directly optimized parameters and transferred parameters for $p = 1, 2, 3$ and varying weight distributions. Markers indicate median values, shaded regions are interquartile range, and lines guide the eye. The median (over the full dataset) decrease in the approximation ratio is only 2.0 p.p. with the transferred parameters. Using KDE reduces this gap to 1.2 p.p., and BFGS reduces it further, as described in the text.

Fig. 5. Gap between the QAOA approximation ratio with transferred ($r_t$) and directly optimized ($r_{opt}$) parameters and the standard deviation of the weights for $p = 1$ and 14-node graphs. If the standard deviation is small, parameter transfer works well, and the gap $r_t - r_{opt}$ is small.

Note that all the statistics at each $p$ are obtained by using a single value of median parameters, irrespective of the graph size and the edge weights distribution.

We observe that the transferred parameters are sufficiently close to the optimized ones that a single local optimization run nearly always recovers an approximation ratio equivalent to the ratio achieved by the directly optimized parameters. Specifically, for each graph we perform one BFGS optimization run using the transferred median parameters as the starting point, and we observe that the optimized approximation ratio is recovered up to high precision ($10^{-7}$) for 96.35% of instances considered in Figure 4. Moreover, in 96.34% of the cases, we were able to use parameter...
symmetries to show that the parameters recovered by the BFGS are the same as the directly optimized ones up to $10^{-4}$. This result indicates that the transferred median parameters are in the same attraction basin as the optimized ones, suggesting that using them as the initial guess can greatly simplify parameter optimization even in the cases where very high-quality QAOA parameters are desired.

The performance of parameter transfer can be understood further by examining how far the transferred parameters are from directly optimized ones. To compute the distances, we first map the optimized parameters to symmetry-related parameters that minimize $||\beta_{\text{opt}} - \beta_{\text{transf}}||$ as described in Appendix A. This gives a unique distance for each instance. The Euclidean distances between directly optimized parameters and transferred medians are given in Figure 6 (left). The distances increase with $p$, partly because of the increase in the dimension of the parameter space in which the distance is computed. As expected from (9), we consistently observe a small distance between directly optimized and transferred $\beta$. For the parameter $\gamma$, we observe that the distance is larger for distributions on which the performance of the transferred parameters is worse. Although the difference varies based on the distributions, the fact that the absolute value of the difference remains small (<0.1 median) for all the distributions in Figure 4 explains the success of BFGS in retrieving the optimized parameters.

Intuitively, it is unlikely that one set of median parameters could be transferred to any instance by a simple rescaling. To test the limits of parameter transfer, we consider weights sampled from a truncated Cauchy distribution. This generates edge weights with varying orders of magnitude, leading to a significantly higher standard deviation of weights than with the other distributions and also greater variations between randomly drawn weight instances, as seen in Figure 5. The performance of parameter transfer on problems with weights drawn from a Cauchy distribution is given in Figure 6 (right). Here, direct parameter transfer from our scheme is less effective, as anticipated in connection with the objective landscape of Figure 3(d). Often, however, the transferred parameters still serve as a good initial guess for local optimization, where they return final approximation ratios equivalent to those from direct optimization in 37.44% of the cases. Hence, the transfer is useful for minimizing the cost of parameter optimization even in this extreme setting.

We observe much larger distances between directly optimized and transferred $\beta$ for the Cauchy weight distribution. However, of all large (>0.1) distances in $\beta$, 99.9% correspond to the cases where the optimized $\gamma$ has both negative and positive components. This is in contrast to the other cases where typically all parameters have the same sign.

To further improve the performance of parameter transfer, we follow Khairy et al. [32] and train a generative model that learns a metadistribution of optimized QAOA parameters using KDE. We train the model using the same dataset of scaled preoptimized parameters for nine-node graphs given by (8), with details in Appendix A. For each graph, we sample 10 parameters ($\beta, \gamma$) from the model and use them in addition to the median parameters, for a total of 11 parameters evaluated, and report the best approximation ratio. At the modest cost of only 10 extra samples, we can significantly reduce the gap between the performance with transferred and directly optimized parameters. The median (over the full dataset) gap is reduced by 40%, from 2.0 p.p. to 1.2 p.p., with KDE reducing the gap over using only the median parameters in 89.5% of the cases. The trends observed for median parameters persist when using the KDE-based method: the performance decreases with depth (0.7 p.p. for $p = 1$, 1.2 p.p. for $p = 2$, and 1.8 p.p. for $p = 3$) and with harder distribution (0.6 p.p. for uniform on $(0, 1)$, 1.0 p.p. for uniform on $(-1, 1)$, and 2.7 p.p. for exponential). This corresponds to a median reduction of the gap by 25% to 50%, with the worst-case median ($p = 3$, exponential distribution) reduced to 3.5 p.p. By construction, the KDE-based technique always gives an approximation ratio better than or equal to using only the median.
DISCUSSION

Our results demonstrate that parameter concentration previously established for unweighted MaxCut [18–21] and the SK model [8] also applies to instances of weighted MaxCut, provided the problem is appropriately rescaled. At each $p$, the parameters from a single QAOA circuit (obtained by taking the median over a large dataset) can be rescaled and transferred across tens of thousands of instances of weighted MaxCut with a performance that is near to that of QAOA with directly optimized parameters. This significantly reduces the time-consuming and difficult step of finding high-quality parameters in QAOA for weighted MaxCut.

The evidence we present considers depth $p \leq 3$. Larger $p$ may be necessary for future applications. However, several recent results show that for unweighted MaxCut, similar parameters work well for most instances even at larger depths [14, 20]. Based on the successes for the $p$ tested and the small changes we observe as $p$ increases, it is likely that this may be true for weighted MaxCut provided the parameters are rescaled to account for weights. Similarly, previous results for unweighted MaxCut [21] and the SK model [7] have shown that parameter concentration persists as $|V|$ increases, consistent with our observations for weighted MaxCut across the $|V|$ we tested. This suggests that parameter concentration in weighted MaxCut may extend to larger $|V|$ and $p$ that are needed for quantum computational advantage in scientifically relevant problems.

The extreme example of Cauchy distributed weights highlights the limitations of our parameter transfer approach specifically and “typical case” results in general. Parameter transfer in a typical case, over a large ensemble of instances, does not necessarily indicate success in interesting practical cases. It remains to be seen whether problems that are relevant to science and industry exhibit properties amenable to parameter transfer and, if not, whether approaches can be developed to overcome these limitations.

APPENDIX

A DETAILS OF THE INSTANCES, PARAMETER OPTIMIZATION, AND ANALYSIS

In our numerical results, we consider weights drawn from differing distributions to probe how QAOA performance depends on the variance of the weights and the presence of negative weights.
We use uniform distributions of positive weights, uniform distributions of positive and negative weights, an exponential distribution of positive weights, and a truncated Cauchy distribution with negative and positive weights:

\[
\begin{align*}
  p_{\text{uniform}^+}(w_{i,j}) &\propto 1, \quad (11) \\
p_{\text{uniform}^\pm}(w_{i,j}) &\propto 1, \quad (12) \\
p_{\exp}(w_{i,j}) &\propto e^{-\alpha w_{i,j}} \quad (13) \\
p_{\text{trunc. Cauchy}}(w_{i,j}) &\propto \frac{1}{1 + w_{i,j}^2}. \quad (14)
\end{align*}
\]

We use rejection sampling to generate samples from the exponential and truncated Cauchy. For the exponential distribution, we use a cutoff \( w_{i,j}^{\exp} \leq 16/\alpha \), and without loss of generality, we set \( \alpha = 1 \). In contrast, an exponential distribution on \([0, \infty]\) would have a probability \( \int_{16/\alpha}^{\infty} e^{-\alpha w} dw = e^{-16} \approx 10^{-7} \) to sample a \( w_{i,j}^{\exp} > 16/\alpha \). For the truncated Cauchy distribution, we generate samples with \(-10^3 \leq w_{i,j} \leq 10^3\). This gives samples with high variance but is distinct from the full Cauchy distribution on \([-\infty, \infty]\).

For our QAOA parameter optimization runs, we use a standard approach based on generating random initial \( \beta \) and \( \gamma \) and then optimizing \( \langle C(\beta, \gamma) \rangle \) with the BFGS algorithm to find a local optimum. We expect our results to not be specific to BFGS and to apply to other choices of local optimizers capable of obtaining high-quality parameters \( \hat{\beta}, \hat{\gamma} \).

For \( |V| = 8 \), we used the BFGS implementation of Press et al. [34]. We compared this with the l-BFGS (low storage BFGS) implementation of NLopt [35–37] in test cases and found that each approach gave satisfactory final results, whereas the approach of Press et al. [34] gave final parameters that would more frequently vary significantly from the initial parameters. For \( n = 14 \) and \( n = 20 \), we used the NLopt implementation, since we found this took less compute time for these instances. The initial parameters were sampled uniformly at random from \(-\pi/4 \leq \beta_l \leq \pi/4 \), which covers the full range of the \( \beta_l \) up to symmetries \([14, 18] \), and with \(-\pi/|w| \leq \gamma_l \leq \pi/|w| \). This focuses on the region around the maximum at small \( \gamma_l \) for all \( l \) because larger ranges tended to produce inferior results. Note that since the small values of \( \gamma_l \) do not always maximize \( \langle C(\gamma, \beta) \rangle \), the optimized parameters are not guaranteed to be globally optimal. For all distributions except Cauchy, for each graph we optimized from 50, 200, and 1,500 random initial parameters at \( p = 1, 2, 3 \), respectively; for the truncated Cauchy distribution, we used 200, 500, and 3,000 seeds, respectively. The final optimized results for each graph were set to be the parameter set with the highest approximation ratio.

To learn the metadistribution of optimized QAOA parameters, we use KDE. We train the model using optimized parameters \( \{x_j = (\beta_j^s, \gamma_j^s)\}_{j=1}^N \) for unweighted MaxCut on all 261,080 nonisomorphic nine-node graphs [18, 24], where the parameters have been scaled according to (8). We use the Parzen–Rosenblatt smooth density estimate with a Gaussian kernel, with probability density function given by

\[
\hat{f}_X(x) = \frac{1}{N} \sum_{i=0}^{N-1} K_\omega(x, x_i^s), \quad (15)
\]

where \( K_\omega(x, x_i^s) = \frac{1}{(2\pi\omega)^n} \exp\left(-\frac{(x-x_i^s)^T(x-x_i^s)}{2\omega^2}\right) \). The kernel bandwidth \( \omega \) is optimized by using a grid search to maximize the log-likelihood of the out-of-sample data using fivefold cross validation. To sample from the density function (15), we sample a uniformly random point \( x_j \) and add to it a sample drawn from a multivariate Gaussian distribution with zero mean and diagonal covariance.
matrix $\omega^2 I$ with the appropriate identity matrix $I$. We use the scikit-learn [38] implementation of KDE. We release the code and the resulting trained KDE model in QAOKit [25].

**Proof $\gamma$ Scales with Objective Value**

**Theorem A.1.** Consider QAOA applied to two objective functions $C_w$ and $C$, $C_w = wC$ with $w > 0$. Let $r$ be the QAOA approximation ratio for $C$ with parameters $(\beta, \gamma)$. Then, QAOA with parameters $(\beta, \gamma/w)$ achieves an identical approximation ratio for $C_w$—that is, $r_w = r$.

**Proof.** Let $C$ and $C_w$ be the diagonal Hamiltonians representing the objective functions $C$ and $C_w$, respectively. From the definition $C_w = wC$,

$$|\beta, \gamma/w, C_w\rangle = \prod_{l=1}^{\rho} e^{-i\beta_l B} e^{-i\gamma_l C} |+\rangle = \prod_{l=1}^{\rho} e^{-i\beta_l B} e^{-i\gamma_l w C} |+\rangle = |\beta, \gamma, C\rangle.$$  

The objectives for the two states are related by

$$\langle C_w(\beta, \gamma/w) \rangle = \langle \beta, \gamma/w, C_w | C_w | \beta, \gamma/w, C_w \rangle = w \langle \beta, \gamma/w, C | \beta, \gamma/w, C \rangle = w \langle C | \beta, \gamma \rangle.$$  

The maximum and minimum objective values for $C$ and $C_w$ are related by the constant factor $w$, $wC_{\text{max}} = (C_{\text{w}})_{\text{max}}$ and $wC_{\text{min}} = (C_{\text{w}})_{\text{min}}$. Thus, the approximation ratios achieved by QAOA applied to $C$ and $C_w$ with parameters $(\beta, \gamma)$ and $(\beta, \gamma/w)$, respectively, are identical:

$$r = \frac{\langle \beta, \gamma, C | C | \beta, \gamma, C \rangle - C_{\text{min}}}{C_{\text{max}} - C_{\text{min}}} = \frac{w \langle \beta, \gamma/w, C_w | C_w | \beta, \gamma/w, C_w \rangle - (C_{\text{w}})_{\text{min}}}{(C_{\text{w}})_{\text{max}} - (C_{\text{w}})_{\text{min}}} = r_w. \quad (18)$$

**A.1 Parameter Symmetries and the Parameter Distance**

QAOA parameters are known to exhibit a variety of symmetries [14, 18] that give identical objective values for symmetry-related parameters. The first symmetry that is pertinent here is time-reversal symmetry, $(\beta, \gamma) \rightarrow (-\beta, -\gamma)$. This is observed, for example, in Figure 3 as a reflection about the origin. The second symmetry is periodicity in $\beta$, $(\beta, \gamma) \rightarrow (\beta \pm \pi/2, \gamma)$.

To define a meaningful angle distance and obtain our results in Figure 6 (left), we must account for these symmetries. The reason is that two identical sets of parameters, up to symmetries, may have a large distance if they are in different symmetry sectors. The distance is then an arbitrary factor, related to symmetry sector that is chosen in each instance rather than to characteristic differences in the parameters themselves. To avoid ambiguities in the distance that arise from the freedom to choose from among symmetry-related parameters, for each optimized $(\beta_{\text{opt}}, \gamma_{\text{opt}})$, we use the preceding symmetries to define a set of parameters $(\beta_{\text{opt}}, \gamma_{\text{opt}})$ that minimizes the distance $||\beta_{\text{opt}} - \beta_{\text{trans}}||$. This is done by considering all combinations of permutations of the component parameters $(\beta_{\text{opt}}) t \rightarrow (\beta_{\text{opt}}) t \pm \pi/2$ and time reversal symmetry on the joint $(\beta_{\text{opt}}, \gamma_{\text{opt}})$. The result is a unique distance that minimizes $||\beta_{\text{opt}} - \beta_{\text{trans}}||$ from among all choices of symmetry-related parameters.

**DISCLAIMER**

This article was prepared for information purposes with contributions from the Future Lab for Applied Research and Engineering (FLARE) Group of JPMorgan Chase & Co. and its affiliates, and is not a product of the Research Department of JPMorgan Chase & Co. JPMorgan Chase & Co. makes no explicit or implied representation and warranty, and accepts no liability, for the completeness, accuracy or reliability of information, or the legal, compliance, tax or accounting effects of matters.
contained herein. This document is not intended as investment research or investment advice, or a recommendation, offer or solicitation for the purchase or sale of any security, financial instrument, financial product or service, or to be used in any way for evaluating the merits of participating in any transaction.

DATA AVAILABILITY
The dataset generated during the current study is available in QAOAKit. The direct URLs to the datafiles are provided in the References section \[30, 31\].

CODE AVAILABILITY
The code that reproduces the figures presented in this article from the published data (direct URL \[33\]) and the code that performs parameter transfer (direct URL \[25\]) are available in QAOAKit.

REFERENCES
[1] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C. Bardin, Rami Barends, Rupak Biswas, et al. 2019. Quantum supremacy using a programmable superconducting processor. Nature 574, 7779 (2019), 505–510. DOI: http://dx.doi.org/10.1038/s41586-019-1666-5
[2] Matthew P. Harrigan, Kevin J. Sung, Matthew Neely, Kevin J. Satzinger, Frank Arute, Kunal Arya, Juan Atalaya, et al. 2021. Quantum approximate optimization of non-planar graph problems on a planar superconducting processor. Nature Physics 17 (2021), 332–336. DOI: http://dx.doi.org/10.1038/s41567-020-01105-y
[3] Yuri Alexeev, Dave Bacon, Kenneth R. Brown, Robert Calderbank, Lincoln D. Carr, Frederic T. Chong, Brian DeMarco, et al. 2021. Quantum computer systems for scientific discovery. PRX Quantum 2, 1 (2021), Article 017001. DOI: http://dx.doi.org/10.1103/prxquantum.2.017001
[4] Tad Hogg and Dmitriy Portnov. 2000. Quantum optimization. Information Sciences 128, 3-4 (2000), 181–197. DOI: http://dx.doi.org/10.1016/s0020-0255(00)00052-9
[5] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. 2014. A quantum approximate optimization algorithm. arXiv:1411.4028 (2014).
[6] Jonathan Wurtz and Peter Love. 2021. Counterdiabaticity and the quantum approximate optimization algorithm. arXiv:2106.15645 (2021).
[7] Sami Boulebnane and Ashley Montanaro. 2021. Predicting parameters for the quantum approximate optimization algorithm for Max-Cut from the infinite-size limit. arXiv:2110.10685 (2021).
[8] E. Farhi, J. Goldstone, S. Gutmann, and L. Zhou. 2020. The quantum approximate optimization algorithm and the Sherrington–Kirkpatrick model at infinite size. arXiv:1910.08187 (2020).
[9] Jonathan Wurtz and Peter Love. 2021. MaxCut quantum approximate optimization algorithm performance guarantees for $p > 1$. Physical Review A 103, 4 (2021), Article 042612. DOI: http://dx.doi.org/10.1103/PhysRevA.103.042612
[10] Danylo Lykov, Roman Schutski, Alexey Galda, Valerii Vinokur, and Yuri Alexeev. 2020. Tensor network quantum simulator with step-dependent parallelization. arXiv:2012.02430 (2020).
[11] Matija Medvidović and Giuseppe Carleo. 2021. Classical variational simulation of the quantum approximate optimization algorithm. NPJ Quantum Information 7 (2021), Article 101. DOI: http://dx.doi.org/10.1038/s41534-021-00440-z
[12] Ruslan Shaydulin, Stuart Hadfield, Tad Hogg, and Ilya Safro. 2021. Classical symmetries and the quantum approximate optimization algorithm. Quantum Information Processing 20, 11 (2021), 1–28. DOI: http://dx.doi.org/10.1007/s11128-021-03298-4
[13] Ruslan Shaydulin and Yuri Alexeev. 2019. Evaluating quantum approximate optimization algorithm: A case study. In Proceedings of the 10th International Green and Sustainable Computing Conference. DOI: http://dx.doi.org/10.1109/IGSC48788.2019.8957201
[14] Leo Zhou, Sheng-Tao Wang, Soonwon Choi, Hannes Pichler, and Mikhail D. Lukin. 2020. Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices. Physical Review X 10 (2020), Article 021067. DOI: http://dx.doi.org/10.1103/PhysRevX.10.021067
[15] Gavin E. Crooks. 2018. Performance of the quantum approximate optimization algorithm on the maximum cut problem. arXiv:1811.08419 (2018).
[16] Ruslan Shaydulin, Ilya Safro, and Jeffrey Larson. 2019. Multistart methods for quantum approximate optimization. In Proceedings of the IEEE High Performance Extreme Computing Conference. 1–8. DOI: http://dx.doi.org/10.1109/hpec.2019.8916288
[17] Ruslan Shaydulin and Stefan M. Wild. 2021. Exploiting symmetry reduces the cost of training QAOA. IEEE Transactions on Quantum Engineering 2 (2021), 1–9. DOI: http://dx.doi.org/10.1109/tqec.2021.3066275

ACM Transactions on Quantum Computing, Vol. 4, No. 3, Article 19. Publication date: April 2023.
Parameter Transfer for Quantum Approximate Optimization of Weighted MaxCut

19:15

[18] Phillip C. Lotshaw, Travis S. Humble, Rebekah Herrman, James Ostrowski, and George Siopsis. 2021. Empirical performance bounds for quantum approximate optimization. Quantum Information Processing 20, 12 (2021), 403. DOI: http://dx.doi.org/10.1007/s11128-021-03342-3

[19] Alexey Galda, Xiaoyuan Liu, Danylo Lykov, Yuri Alexeev, and Ilya Safro. 2021. Transferability of optimal QAOA parameters between random graphs. arXiv:2106.07531 (2021).

[20] Jonathan Wurtz and Danylo Lykov. 2021. The fixed angle conjecture for QAOA on regular MaxCut graphs. arXiv:2107.00677 (2021).

[21] Fernando G. S. L. Brandão, Michael Broughton, Edward Farhi, Sam Gutmann, and Hartmut Neven. 2018. For fixed control parameters the quantum approximate optimization algorithm’s objective function value concentrates for typical instances. arXiv:1812.04170 (2018).

[22] Andrew Lucas. 2014. Ising formulations of many NP problems. Frontiers in Physics 2 (2014), 5. DOI: http://dx.doi.org/10.3389/fphy.2014.00005

[23] M. E. J. Newman. 2006. Modularity and community structure in networks. Proceedings of the National Academy of Sciences 103 (2006), 8577. DOI: http://dx.doi.org/10.1073/pnas.0601602103

[24] Ruslan Shaydulin, Kunal Marwaha, Jonathan Wurtz, and Phillip C. Lotshaw. 2021. QAOAKit: A toolkit for reproducible study, application, and verification of QAOA. In Proceedings of the 2nd International Workshop on Quantum Computing Software. DOI: http://dx.doi.org/10.1109/QCS54837.2021.00011

[25] GitHub. 2022. QAOAKit. Retrieved February 25, 2023 from https://github.com/QAOAKit/QAOAKit/blob/7bdedee1b4ca2a3ea1e6fbd6b228e89e6f5f2/examples/weighted_transfer.ipynb. (2022).

[26] Stuart Hadfield. 2018. Quantum Algorithms for Scientific Computing and Approximate Optimization. Ph.D. Dissertation. Columbia University.

[27] Asier Ozaeta, Wil van Dam, and Peter L. McMahon. 2021. Expectation values from the single-layer quantum approximate optimization algorithm on Ising problems. arXiv:2012.03421v2 (2021).

[28] Zhihui Wang, Stuart Hadfield, Zhang Jiang, and Eleanor G. Rieffel. 2018. Quantum approximate optimization algorithm for MaxCut: A fermionic view. Physical Review A 97, 2 (2018), Article 022304. DOI: http://dx.doi.org/10.1103/physreva.97.022304

[29] Joao Basso, Edward Farhi, Kunal Marwaha, Benjamin Villalonga, and Leo Zhou. 2021. The quantum approximate optimization algorithm at high depth for MaxCut on large-girth regular graphs and the Sherrington–Kirkpatrick model. arXiv:2110.14206 (2021).

[30] GitHub. 2022. QAOAKit/data. Retrieved February 25, 2023 from https://github.com/QAOAKit/data/blob/bdd9220486b456ce93727d75ed9308a4fc8a9c8/transfer_qaoa_weighted/transfer_qaoa_weighted.zip. (2022).

[31] GitHub. 2022. QAOAKit/data. Retrieved February 25, 2023 from https://github.com/QAOAKit/data/blob/bdd9220486b456ce93727d75ed9308a4fc8a9c8bfgs.zip. (2022).

[32] Sami Khairy, Ruslan Shaydulin, Lukasz Cincio, Yuri Alexeev, and Prasanna Balaprakash. 2020. Learning to optimize variational quantum circuits to solve combinatorial problems. Proceedings of the AAAI Conference on Artificial Intelligence 34, 03 (2020), 2367–2375. DOI: http://dx.doi.org/10.1609/aaai.v34i03.5616

[33] GitHub. 2022. QAOAKit/examples. Retrieved February 25, 2023 from https://github.com/QAOAKit/QAOAKit/blob/7bede1b4ca2a3ea1e6fbd6b228e89e6f5f2/examples/reproduce_all_figures_from_Parameter_Transfer_for_Quantum_Approximate_Optimization_of_Weighted_MaxCut.ipynb. (2022).

[34] William H. Press, Brian P. Flannery, and Saul A. Teukolsky. 1993. Numerical Recipes in Fortran 77: The Art of Scientific Computing (2nd ed.). Cambridge University Press.

[35] Steven G. Johnson. 2022. The NLopt Nonlinear-Optimization Package. Retrieved February 25, 2023 from http://github.com/stevengj/nlopt.

[36] Jorge Nocedal. 1980. Updating quasi-Newton matrices with limited storage. Mathematics of Computation 35 (1980), 773–782. DOI: http://dx.doi.org/10.1090/S0025-5718-1980-0572855-7

[37] Dong C. Liu and Jorge Nocedal. 1989. On the limited memory BFGS method for large scale optimization. Mathematical Programming 45 (1989), 503–528. DOI: http://dx.doi.org/10.1007/BF01589116

[38] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, et al. 2011. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12 (2011), 2825–2830.

Received 1 July 2022; revised 9 January 2023; accepted 9 February 2023