

Parameters Fixing Strategy for Quantum Approximate Optimization Algorithm

Xinwei Lee

*Graduate School of Systems and Information Engineering
University of Tsukuba
Tsukuba, Ibaraki, JAPAN
xwlee@cavelab.cs.tsukuba.ac.jp*

Yoshiyuki Saito

*Graduate School of Computer Science and Engineering
University of Aizu
Aizu-Wakamatsu, Fukushima, JAPAN
m5241141@u-aizu.ac.jp*

Dongsheng Cai

*Faculty of Engineering, Information and Systems
University of Tsukuba
Tsukuba, Ibaraki, JAPAN
cai@cs.tsukuba.ac.jp*

Nobuyoshi Asai

*School of Computer Science and Engineering
University of Aizu
Aizu-Wakamatsu, Fukushima, JAPAN
nasai@u-aizu.ac.jp*

Abstract—The quantum approximate optimization algorithm (QAOA) has numerous promising applications in solving the combinatorial optimization problems on near-term Noisy Intermediate Scalable Quantum (NISQ) devices. QAOA has a quantum-classical hybrid structure. Its quantum part consists of a parameterized alternating operator ansatz, and its classical part comprises an optimization algorithm, which optimizes the parameters to maximize the expectation value of the problem Hamiltonian. This expectation value depends highly on the parameters, this implies that a set of good parameters leads to an accurate solution. However, at large circuit depth of QAOA, it is difficult to achieve global optimization due to the multiple occurrences of local minima or maxima. In this paper, we propose a parameters fixing strategy which gives high approximation ratio on average, even at large circuit depths, by initializing QAOA with the optimal parameters obtained from the previous depths. We test our strategy on the Max-cut problem of certain classes of graphs such as the 3-regular graphs and the Erdős-Rényi graphs.

I. INTRODUCTION

Since the introduction of QAOA by Farhi et al. [1], it is widely known for its efficiency and universality in solving combinatorial optimization problems on quantum computers [2], [3]. Among the problems, the Max-cut problem is heavily studied for its simple formulation and its deep relation to the Ising model [4]. There exist some classes of graphs which the Max-cut problem can be solved analytically, such as the bipartite graphs, the 2-regular (ring) graphs and the fully-connected graphs. However, for the other classes of graphs there is no analytical solution, and they are NP-hard [5].

Although QAOA does not give the exact solution to the Max-cut problem, the algorithm provides a heuristic approach for the problem. For classical heuristics, the Goemans-Williamson algorithm (widely known as GW algorithm) is able to achieve the approximation ratio of at least 0.878 on

the Max-cut problem through semidefinite programming [6]. Farhi et al. stated in [1] that QAOA is able to achieve the approximation ratio of at least 0.6924 on 3-regular graphs for $p = 1$, where p is the depth of the quantum circuit for QAOA. Recently, Wurtz et al. [7] extends the result to $p > 1$, with the lower bound of the approximation ratio as 0.7559 for $p = 2$ and 0.7924 for $p = 3$, for 3-regular graphs. It is obvious that the lower bound for the approximation ratio increases as p increase, and will surpass the lower bound for the GW algorithm for large enough p . This agrees with [1] that as $p \rightarrow \infty$, the approximation ratio obtainable from QAOA approaches 1, in which we obtain the true solution to the problem.

However, several hurdles continue to exist in the practical application of QAOA. One of them is obtaining a “good” solution at large circuit depths. Solving large problems (graphs with large number of nodes) often requires deep circuits. However, due to the existence of local maxima on the hypersurface of the expectation function, local optimizers tend to be trapped inside one of the local maximum, whilst the desired solution are the global maxima. Many researches, for example [8]–[16], have stated this problem as the limitation of QAOA at large circuit depths. Some researches [12]–[18] discussed the strategies to improve the performance of QAOA, which will be discussed later. It is now clear that the performance of QAOA heavily depends on its initial parameters, whether it converges to a local maximum or a global maximum, and hence its performance. This motivates our work to introduce a strategy that aims to choose the QAOA parameters such that they yield better performance overall.

In this paper, we discuss a straightforward, yet practically effective, parameters fixing strategy to improve the average performance of QAOA at large circuit depths. First, using simulation results, we show that with random initial angles,

QAOA will have a poor performance on average at large circuit depths and large problems. With our strategy applied, the average performances on the same problems are improved. We choose our problems from the widely studied 3-regular graphs and the Erdős-Rényi random graphs to verify the capability of our strategy in solving those problem instances.

II. QAOA FOR MAX-CUT

QAOA originates from the Quantum Adiabatic Algorithm (QAA) [19], which focuses on evolving the initial Hamiltonian H_B to the problem Hamiltonian H_C , satisfying

$$\tilde{H}(s) = (1-s)H_B + sH_C, \quad (1)$$

where $s(t) \rightarrow 1$ as $t \rightarrow \infty$. The evolution in Eq. (1) is then discretized, which results in QAOA. In QAOA, the alternating unitary operators involving H_B and H_C are applied to the initial state to simulate the evolution of the system in Eq. (1):

$$|\psi_p(\vec{\gamma}, \vec{\beta})\rangle = e^{-i\beta_p H_B} e^{-i\gamma_p H_C} \dots e^{-i\beta_1 H_B} e^{-i\gamma_1 H_C} |+\rangle^{\otimes n}, \quad (2)$$

where $\vec{\gamma} = (\gamma_1, \gamma_2, \dots, \gamma_p)$ and $\vec{\beta} = (\beta_1, \beta_2, \dots, \beta_p)$ are the $2p$ variational parameters, with $\gamma_i \in [0, 2\pi)$ and $\beta_i \in [0, \pi)$. $|+\rangle^{\otimes n}$ corresponds to n qubits in the ground state of the Pauli- X basis. For the Max-cut problem of a graph $G = (V, E)$, H_C and H_B are given as

$$H_C = \frac{1}{2} \sum_{(j,k) \in E} (I - Z_j Z_k), \quad (3)$$

$$H_B = \sum_{j \in V} X_j. \quad (4)$$

X_j and Z_j are the Pauli operators acting on the j -th qubit. After applying the operators as in Eq. (2), we calculate the expectation of the operator H_C with respect to the ansatz state $|\psi_p(\vec{\gamma}, \vec{\beta})\rangle$:

$$F_p(\vec{\gamma}, \vec{\beta}) = \langle \psi_p(\vec{\gamma}, \vec{\beta}) | H_C | \psi_p(\vec{\gamma}, \vec{\beta}) \rangle. \quad (5)$$

Since Eq. (5) is parameterized by $\vec{\gamma}$ and $\vec{\beta}$, we can use a classical optimization algorithm to search for the angles which maximize F_p :

$$(\vec{\gamma}^*, \vec{\beta}^*) = \arg \max_{\vec{\gamma}, \vec{\beta}} F_p(\vec{\gamma}, \vec{\beta}). \quad (6)$$

Fig. 1 shows the schematic diagram of such a routine for QAOA. First, some initial parameters $(\vec{\gamma}, \vec{\beta})$ is passed into the quantum circuit of QAOA, which is characterized by Eq. (2), (3) and (4). The results of the measurement are then used to calculate F_p from (5), and the value of F_p is optimized using a classical optimizer to obtain a better $(\vec{\gamma}, \vec{\beta})$ to be passed into the quantum circuit again. This continues until the optimizer converges and the optimal parameters $(\vec{\gamma}^*, \vec{\beta}^*)$ is obtained. The *approximation ratio* α is defined as

$$\alpha = \frac{F_p(\vec{\gamma}^*, \vec{\beta}^*)}{C_{\max}}, \quad (7)$$

where C_{\max} is the maximum cut value for the graph. The approximation ratio indicates how near the solution given by

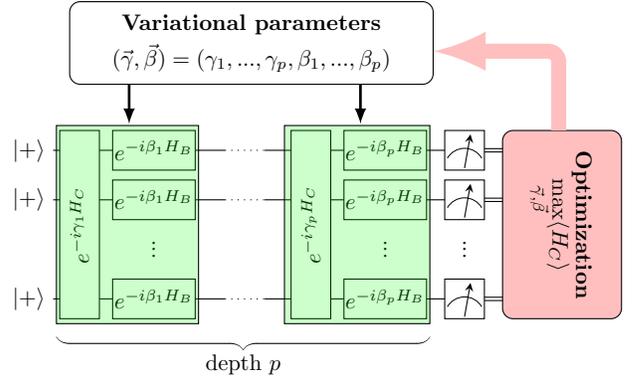


Fig. 1: The routine of the Quantum Approximate Optimization Algorithm (QAOA).

QAOA is to the true solution, ranging from 0 to 1, with 0 the furthest and 1 the nearest to the true solution. It will be the performance measure throughout this paper as we aim to approximate the solution of the Max-cut problem as accurate as possible.

As the circuit depth p increases, $\max F_p(\vec{\gamma}, \vec{\beta})$ also increases, and it will approach C_{\max} as $p \rightarrow \infty$, thus the approximation ratio α will approach 1 [1]. However, according to our numerical experiments, the optimizers does not necessarily succeed in finding the maximum value of F_p as p gets larger. Apparently, which parameters are chosen as the initial points to the optimizer plays an important role in whether the optimizer converges to a global maximum. Hence, we would prefer to have “good” initial points in order for the optimizer to converge to the global maximum.

III. RELATED WORK

While many strategies of choosing good parameters for QAOA are introduced and discussed, there is no strong evidence that which strategy is more effective than the others. Some [12], [13], [15], [20] found out that there is a linear relationship between the parameters and the circuit depths, which resembles the linear annealing scheme. As shown in their work, the optimal parameters γ_i increases linearly and β_i decreases linearly, where i is the index of the parameters in $(\vec{\gamma}, \vec{\beta})$. However, besides the linearly increasing and decreasing parameters, the landscape of the expectation function also has multiple maximum points, which arguments are some sets of parameters that do not follow the linear pattern. In fact, in most cases of our simulations, the optimal parameters we obtain do not follow the linear pattern.

Brandao et al. [17] found out that for the Max-cut problem, the parameters for the typical instances of 3-regular graphs concentrate. This results in similar expectation function for different 3-regular graphs regardless the number of nodes, as long as the structure of the graph is the same. They leverage this property to reuse the optimal parameters obtained from optimizing smaller graphs in the optimization of larger graphs, which they call it the leapfrogging strategy. They state that this allows them to find good solutions using QAOA with fewer

calls to the quantum computer. Their strategy also applies to Erdős-Rényi graphs with edge probability $3/(n-1)$, as they have the expected graph degree of 3.

Zhou et al. [12] introduces two strategies for optimizing the variational parameters. First, they discover the linear relationship between the parameters and the circuit depth p in solving the Max-cut problem on 16-vertex unweighted 3-regular graphs using QAOA. They use the linear interpolation method to extend the optimal parameters at depth p to make a guess about the parameters at $p+1$. They call this the INTERP heuristic strategy. The second strategy is the FOURIER heuristic strategy, which uses the Discrete Sine/Cosine Transform to determine the parameters at $p+1$ from the Fourier amplitudes at p . They test the two strategies against the random initialization of the parameters on 40 instances of 16-vertex unweighted and weighted 3-regular graphs, and their strategies outperform the random initialization QAOA after $p \geq 5$.

Shaydulin et al. [16] uses a multistart approach to improve the performance of QAOA on graph clustering problems. They combine two optimization algorithms, namely APOSMM [21], [22] and BOBYQA [23]. APOSMM is an optimization algorithm which coordinates multiple local optimization runs to identify better local optima. They embed the classical local optimizer BOBYQA in APOSMM, which they call the APOSMM+BOBYQA, to improve the QAOA performance. They apply this strategy to solve the modularity maximization community detection problem with graphs between 10 and 12 vertices. They have shown, using simulations, that the APOSMM+BOBYQA method performs better than typical optimizers such as COBYLA [24], NELDER-MEAD [25], and also the BOBYQA alone. They also apply the aforementioned leapfrogging strategy in [17] to reuse optimal parameters as initial points for different graphs.

A more recent work by Sack et al. [15] introduce the initialization based on the Trotterized quantum annealing (TQA) protocol to prevent the convergence to false local minima. This parameter initialization method is inspired by the connection between simulated quantum annealing and QAOA. They suggest the initialization of the i -th parameter in depth p as $\gamma_i = (i/p)\Delta t$ and $\beta_i = (1 - i/p)\Delta t$, which is derived from the discretization of quantum annealing protocol using Suzuki-Trotter decomposition. They also found out that there exist optimal time steps Δt which helps in the convergence to a minima that is very close to the global minima. They apply their strategy to solve the Max-cut problem for 12-vertices over 50 random graphs. Others like [18] use machine learning techniques to predict the initial parameters for higher depth QAOA. After the models are trained, they demonstrate an average improvement of 44.9% in run-time across various local optimization algorithms such as L-BFGS-B, NELDER-MEAD, SLSQP and COBYLA.

IV. NUMERICAL EXPERIMENT

In this paper, the performances of two initialization methods of QAOA are compared: the random initialization and the parameters fixing strategy. We use QAOA simulation

to solve the Max-cut problem on the 3-regular graphs and the Erdős-Rényi random graphs with edge probability of 0.5 with different problem sizes (number of nodes). The simulation of the quantum circuits in QAOA is done by using the Qiskit Aer simulator and the problem graphs are generated using the NetworkX package. We solve the Max-cut problem for the 3-regular graphs with the number of nodes $n = 6, 8, 10, 12, 16$. We also solve for the Erdős-Rényi graphs with $n = 6, 7, 8, 9, 10$. The graph instances are chosen randomly so that they do not have biased structures.

With the parameters initialization methods aside, our common simulation procedure is as follows: first, we pass a set of initial parameters $(\vec{\gamma}_{\text{init}}, \vec{\beta}_{\text{init}})$ into the quantum circuit shown in Fig. 1. We then measure the quantum circuit to obtain the probability distribution of the ansatz state. We calculate the expectation value as

$$\begin{aligned} F(\vec{\gamma}, \vec{\beta}) &= \langle \psi(\vec{\gamma}, \vec{\beta}) | H_C | \psi(\vec{\gamma}, \vec{\beta}) \rangle \\ &= \sum_i \lambda_i p_i, \end{aligned} \quad (8)$$

where λ_i is the eigenvalue (cut-value) of the problem Hamiltonian H_C , and p_i is the probability of measuring the eigenstate correspond to λ_i . The expectation value is then passed into the optimizer for a better set of parameters. This cycle continues until the optimizer terminates and we have the maximized expectation $F(\vec{\gamma}^*, \vec{\beta}^*)$ with its optimal parameters $(\vec{\gamma}^*, \vec{\beta}^*)$. Hence, we can calculate the approximation ratio α for this set of parameters from Eq. (7). In all of our simulations, we use the NELDER-MEAD [25] optimizer provided by the SciPy package, with the settings of 1000 maximum allowed function evaluations and the absolute error tolerance of 10^{-4} . As for the metric to evaluate the performances of the initialization methods, we use the *mean approximation ratio* obtained from the QAOA simulations initialized by different initial parameters, for a fixed QAOA circuit depth p . Additionally, we investigate how the distribution of the approximation ratios changes with p .

A. Random Initialization Method

For the random initialization method, 20 sets of random parameters $(\vec{\gamma}, \vec{\beta})$ are passed into QAOA for each p and we take the average of the 20 approximation ratios output by QAOA. Then, we increase p and repeat. The same procedure is done to different graph instances. This is the most naive method used when solving problems using QAOA. The output data will be the reference to which after our strategy is applied. Fig. 2(a) and (b) show the results of the QAOA simulations using random initialization method. We can observe that for 3-regular graphs, the mean approximation ratio given by 20 random initial parameters does not increase with the circuit depth p overall. For small number of nodes like $n = 6$, the mean approximation ratio increases with p and is able to achieve mean $\alpha > 0.95$ for $p \geq 7$. However, as the number of nodes increases ($n \geq 8$), the mean α only increase for small depths. As the depth gets larger ($p \geq 5$), there is a decrease in the mean α and most of them do not achieve $\alpha > 0.95$

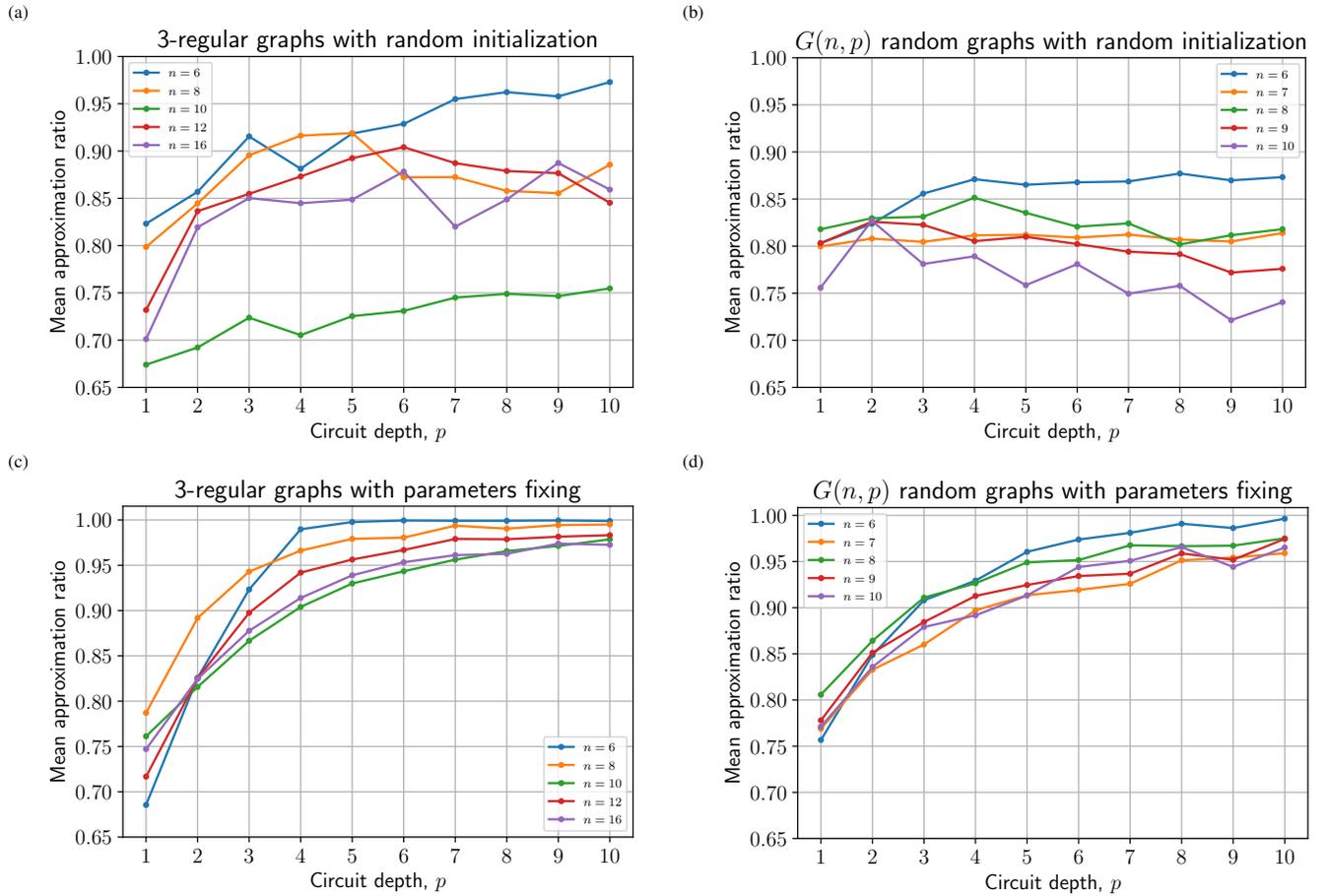


Fig. 2: The comparison of the results produced by applying different parameters initialization methods. The Max-cut instances solved are the 3-regular graphs ($n = 6, 8, 10, 12, 16$) and the Erdős-Rényi (or sometimes $G(n, p)$) graphs ($n = 6, 7, 8, 9, 10$) with edge probability of 0.5. The variation of the mean approximation ratio against the circuit depth is plotted. (a) 3-regular graphs with random initialization, (b) Erdős-Rényi graphs with random initialization, (c) 3-regular graphs with parameters fixing strategy, (d) Erdős-Rényi graphs with parameters fixing strategy.

like the 6-node graphs did, even the depth is increased until $p = 10$. For some n , the standard deviations of α increase as p increases. This implies that at larger p , there exist some initial parameters which converged to our desired real maxima, giving high approximation ratio. On the other hand, there are also some initial parameters which converged to some false local maxima, giving low approximation ratio which dragged down the mean α . For the Erdős-Rényi graphs, similar trend is observed for $n \geq 7$.

B. Parameters Fixing Strategy

At large circuit depth of QAOA, the optimizer does not necessarily find the desired global maxima. Inspired by the linear interpolation strategies in [12], [13], we attempt to make better choices of parameters by extending them from depth p to $p+1$. Starting from $p = 1$, we solve the problem with 20 sets of random parameters. As the optimizer behaves well in shallow depth of QAOA, we entrust it to find our desired solution at $p = 1$. We then select the parameters with the highest approximation ratio and treat it as the optimal parameters for $p = 1$: (γ_1^*, β_1^*) . We fix this set of parameters and insert another pair of random (γ_2, β_2) and pass $(\gamma_1^*, \gamma_2, \beta_1^*, \beta_2)$ as

the initial parameters into solving $p = 2$. Again, to make sure we find the true global maxima, instead of using one random pair of (γ_i, β_i) , we use 20 random pairs of (γ_i, β_i) and select the one which gives us the highest approximation ratio. Hence, we obtain the optimal parameters at $p = 2$ as $(\gamma_1^*, \gamma_2^*, \beta_1^*, \beta_2^*)$. This procedure is repeated until $p = 10$. This is summarized in Algorithm 1. We call this the *parameters fixing* strategy.

Fig. 2(c) and (d) shows the results of the QAOA simulation with the parameters fixing strategy applied. The increasing trend of the mean approximation ratio against the circuit depth can be observed for both the 3-regular graphs and the Erdős-Rényi graphs. The 3-regular instances are able to achieve mean $\alpha > 0.95$ at $p \geq 7$ and the Erdős-Rényi instances achieves mean $\alpha > 0.95$ at $p \geq 8$. This is in contrast to the random initialization method, which the mean α does not increase with increasing p . Since the mean α given by fixing parameters is relatively high, this means that the optimizer rarely converges to a false maxima. By starting QAOA with the optimal parameters from shallower depths, the problem of QAOA converging to a false local minima can thus be subdued, as shown in the results. Hence,

Algorithm 1 Parameters Fixing

```

1: for  $q = 1 \dots p$  do                                ▷ Circuit depth  $p$ 
2:   for  $k = 1 \dots n$  do                                ▷ No. of trials  $n$ 
3:      $\gamma_q \leftarrow \text{rand}(0, 2\pi)$ 
4:      $\beta_q \leftarrow \text{rand}(0, \pi)$ 
5:     if  $q = 1$  then
6:        $(\vec{\gamma}, \vec{\beta}) \leftarrow (\gamma_1, \beta_1)$ 
7:     else
8:        $(\vec{\gamma}, \vec{\beta}) \leftarrow (\gamma_1^*, \dots, \gamma_{q-1}^*, \gamma_q, \beta_1^*, \dots, \beta_{q-1}^*, \beta_q)$ 
          ▷ New parameters  $(\gamma_q, \beta_q)$  are inserted.
9:     end if
10:    Initialize QAOA with  $(\vec{\gamma}, \vec{\beta})$  and optimize to find
       $F_q(\vec{\gamma}^*, \vec{\beta}^*)_k$ .
11:   end for
12:    $F_q(\vec{\gamma}^*, \vec{\beta}^*) \leftarrow \max_k F_q(\vec{\gamma}^*, \vec{\beta}^*)_k$ 
13:    $(\vec{\gamma}^*, \vec{\beta}^*) \leftarrow \arg \max_k F_q(\vec{\gamma}^*, \vec{\beta}^*)_k$ 
          ▷  $(\vec{\gamma}^*, \vec{\beta}^*) = (\gamma_1^*, \dots, \gamma_q^*, \beta_1^*, \dots, \beta_q^*)$ 
14: end for
15: Output:  $(\vec{\gamma}^*, \vec{\beta}^*)$  and  $F_p(\vec{\gamma}^*, \vec{\beta}^*)$ .

```

we can conclude that this method is effective in obtaining good results in QAOA, at least true for the graph instances used in our work. Moreover, we also found that for the 3-regular graphs with parameters fixing, the standard deviation of the α 's decreases with increasing p as shown in Fig. 3. This implies that as p increases, the significance of the new random parameters (γ_i, β_i) becomes less, as no matter what the random value is, the value of α does not change much. The dependence of α is stronger in those optimal parameters from the shallower depths: $(\gamma_1^*, \dots, \gamma_{p-1}^*, \beta_1^*, \dots, \beta_{p-1}^*)$. However, we do not observe this pattern in the results of the Erdős-Rényi graphs.

Although we call this the parameters fixing strategy, in our actual simulation, we allow the optimal parameters $(\gamma_1^*, \dots, \gamma_{p-1}^*, \beta_1^*, \dots, \beta_{p-1}^*)$ to be further optimized as they are passed into QAOA of depth p as the initial parameters. Hence, the optimal parameters at p will be $(\gamma_1^{**}, \dots, \gamma_{p-1}^{**}, \gamma_p^*, \beta_1^{**}, \dots, \beta_{p-1}^{**}, \beta_p^*)$, where the optimal values at $p-1$: (γ_i^*, β_i^*) , is different from the optimal values at p : $(\gamma_i^{**}, \beta_i^{**})$. However, our simulation data in Fig. 4 shows that this difference is actually subtle. As p increases, the optimal parameter at the i -th position stays almost the same relative to other optimal parameters. It is also seen that the parameters tend to change more when they are first introduced, e.g. γ_6 at $p=6$, and change less as the depth increases. This shows that the optimal parameters at p stay optimum at $p+1$. If this was not the case, the values of the optimal parameters at p would have gone through more drastic changes in their values.

V. OPTIMIZATION LANDSCAPE

The optimization landscape (the landscape of the expectation function $F_p(\vec{\gamma}, \vec{\beta})$) is the landscape traversed by the optimizer. It is a crucial part in QAOA as its properties decide the convergence of the local optimizers as mentioned in the

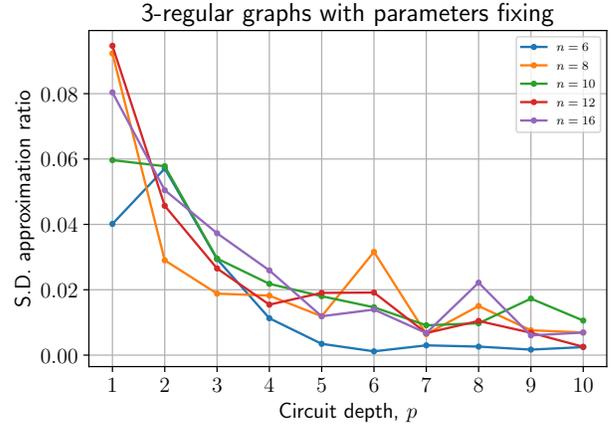


Fig. 3: The variation of the standard deviation of the approximation ratio α against the circuit depth p , for the 3-regular graphs. This is the same set of approximation ratio as in Fig. 2(c).

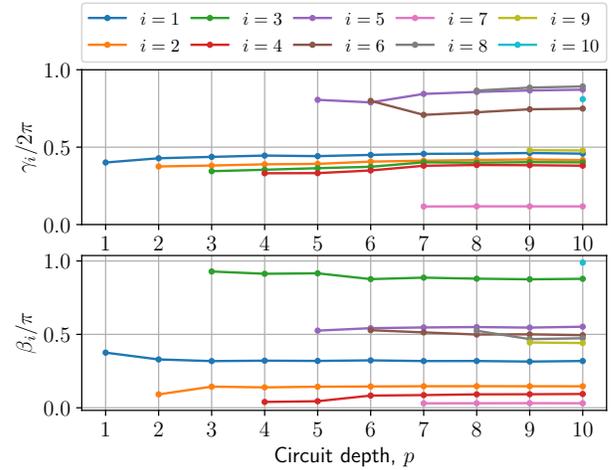


Fig. 4: Changes in the i -th position optimal parameters as they are passed into the next depth of QAOA as the initial parameters. This is extracted from the simulation data of a 8-node 3-regular graph.

previous sections. We investigate the optimization landscape to find out the reason behind the effectiveness of the parameters fixing strategy. We find out that as we fix different points in depth p , the optimization landscape of depth $p+1$ is different, depending on which points are fixed. Fig. 5(a) shows different effects of fixing the parameters of the maximum points (high expectation value) and fixing the parameters of the minimum points (low expectation value) as the depth increases. When the parameters of the maximum points are fixed, the optimization landscape is covered with more red region, which correspond to high expectation values. On the other hand, when we fix the parameters of the minimum points, the optimization landscape is covered with more blue region, which correspond to low expectation values. Additionally, to observe the effect of parameters fixing at large depths, we extract the optimal parameters at $p=10$ from our simulation data and plot the optimization landscape they span. Fig. 5(b) shows the landscape after parameters fixing is applied continuously until

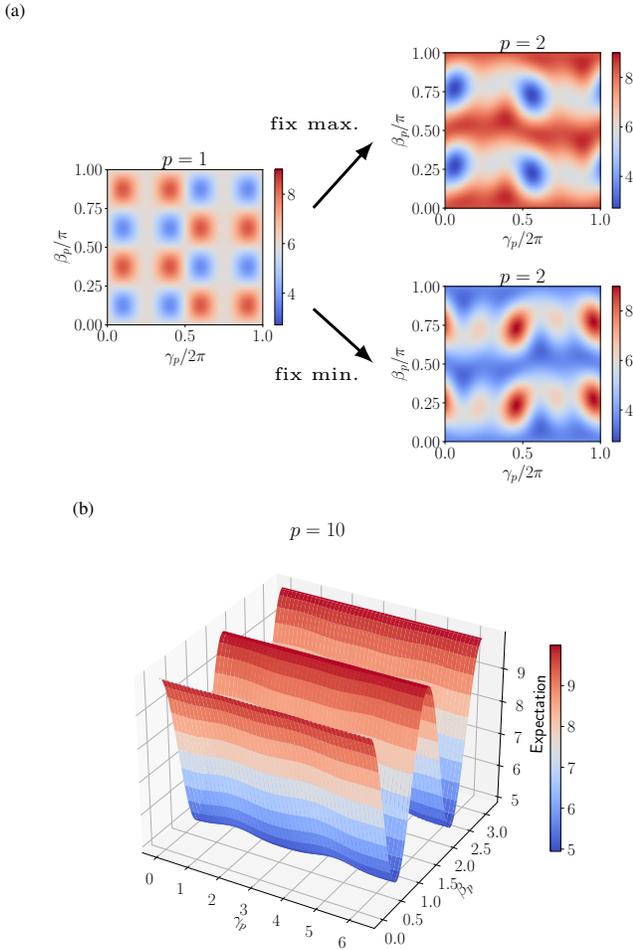


Fig. 5: (a) Heatmap showing the changes in the optimization landscape when different parameters are fixed while the depth increases. Red regions correspond to high expectation values and blue regions correspond to low expectation values. On the left is the heatmap of the optimization landscape for $p = 1$ in the (γ_1, β_1) space. On the right is the heatmap of $p = 2$ in the (γ_2, β_2) space, while (γ_1, β_1) is fixed. (b) The landscape of $p = 10$ with parameters fixing in the $(\gamma_{10}, \beta_{10})$ space. The optimal parameters up until $p = 9$ are extracted from real simulation data. These are the optimization landscapes of the 8-node 3-regular graph used in our simulation.

$p = 10$. The maximum points in the landscape slowly transforms into maximum lines with increasing depth while fixing the optimal parameters. We observe similar pattern in the transformation of landscapes in other graph instances as well. We hypothesize that the maximum lines causes the optimizer becomes less likely to be trapped inside a local maxima. Hence, the approximation ratio is higher on average with the parameters fixing strategy applied.

VI. DISCUSSION AND CONCLUSION

We have introduced the parameters fixing strategy as an initialization method for QAOA and have shown its effectiveness in practice. The strategy reuses optimal parameters from the previous depths to obtain high approximation ratio on average at large QAOA circuit depths, which has high practical importance. We show the effectiveness of this strategy using QAOA simulations to solve the Max-cut problem for the 3-regular

graphs and the Erdős-Rényi graphs, and compare the results with those obtained from random initialization. It is shown that with this strategy applied, the average approximation ratios are higher than those which are initialized randomly. For all the 3-regular instances we considered, the mean approximation ratio $\alpha > 0.95$ is achieved when $p \geq 7$. On the other hand, the Erdős-Rényi instances achieves mean $\alpha > 0.95$ when $p \geq 8$. Our results also show that for 3-regular graphs, the significance of new parameters introduced at higher depths becomes less when parameters fixing is applied. This implies that if the parameters up until depth p are optimal, no matter what new values we choose for $p + 1$, the value of α does not change much. Also, we have shown how the optimization landscape changes as the circuit depth increases during parameters fixing, transforming from maximum points to maximum lines as p increases.

Although the parameters fixing strategy shows good performance in yielding high approximation ratios, it also has apparent drawbacks. As shown in Algorithm 1, this strategy requires $\mathcal{O}(np)$ time, where n is the number of trials to obtain the maximum expectation, and p is the circuit depth for QAOA to be solved. This strategy forces the user to start solving QAOA from small depths like $p = 1$. However, some has proposed the analytical solution for QAOA at small depths for the 3-regular graphs [7], [26]. Therefore, it is possible to obtain the optimal parameters for small depths analytically. Also, the performance of our strategy is tested only with the NELDER-MEAD optimizer. The performances of the other optimizers are yet to be studied.

It is inevitable to solve QAOA at large circuit depths, as large problem instances usually requires the circuit depth to be increased to obtain a good approximation to the solution. Since local optimizers perform badly at large circuit depths, a workaround is needed for the optimizers to converge to the desired global maxima. Although more rigorous proofs are required, the strategy we propose is indeed practically useful, as we have demonstrated its performance using simulation results. We aim to obtain more rigorous proofs for the parameters fixing strategy, and extend our results to other classes of graphs, or even other problems that can be solved using QAOA.

REFERENCES

- [1] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm," 2014.
- [2] S. Lloyd, "Quantum approximate optimization is computationally universal," 2018.
- [3] E. Farhi and A. W. Harrow, "Quantum supremacy through the quantum approximate optimization algorithm," 2019.
- [4] J. C. A. D'Auriac, M. Preissmann, and A. Sebö, "Optimal cuts in graphs and statistical mechanics," *Math. Comput. Model.*, vol. 26, no. 8–10, p. 1–11, Oct. 1997. [Online]. Available: [https://doi.org/10.1016/S0895-7177\(97\)00195-7](https://doi.org/10.1016/S0895-7177(97)00195-7)
- [5] R. Karp, "Reducibility among combinatorial problems," in *Complexity of Computer Computations*, R. Miller and J. Thatcher, Eds. Plenum Press, 1972, pp. 85–103.
- [6] M. X. Goemans and D. P. Williamson, "Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming," *J. ACM*, vol. 42, no. 6, p. 1115–1145, Nov. 1995. [Online]. Available: <https://doi.org/10.1145/227683.227684>

- [7] J. Wurtz and P. Love, “Maxcut quantum approximate optimization algorithm performance guarantees for $p > 1$,” *Physical Review A*, vol. 103, no. 4, Apr 2021. [Online]. Available: <http://dx.doi.org/10.1103/PhysRevA.103.042612>
- [8] G. G. Guerreschi and A. Y. Matsuura, “Qaoa for max-cut requires hundreds of qubits for quantum speed-up,” *Scientific Reports*, vol. 9, no. 1, May 2019. [Online]. Available: <http://dx.doi.org/10.1038/s41598-019-43176-9>
- [9] M. Y. Niu, S. Lu, and I. L. Chuang, “Optimizing qaoa: Success probability and runtime dependence on circuit depth,” 2019.
- [10] M. Willsch, D. Willsch, F. Jin, H. De Raedt, and K. Michielsen, “Benchmarking the quantum approximate optimization algorithm,” *Quantum Information Processing*, vol. 19, no. 7, Jun 2020. [Online]. Available: <http://dx.doi.org/10.1007/s11128-020-02692-8>
- [11] C. Moussa, H. Calandra, and V. Dunjko, “To quantum or not to quantum: towards algorithm selection in near-term quantum optimization,” *Quantum Science and Technology*, vol. 5, no. 4, p. 044009, Oct 2020. [Online]. Available: <http://dx.doi.org/10.1088/2058-9565/abb8e5>
- [12] L. Zhou, S.-T. Wang, S. Choi, H. Pichler, and M. D. Lukin, “Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices,” *Phys. Rev. X*, vol. 10, p. 021067, Jun 2020. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevX.10.021067>
- [13] J. Cook, S. Eidenbenz, and A. Bärtshi, “The quantum alternating operator ansatz on maximum k-vertex cover,” *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 83–92, 2020.
- [14] D. Liang, L. Li, and S. Leichenauer, “Investigating quantum approximate optimization algorithms under bang-bang protocols,” *Phys. Rev. Research*, vol. 2, p. 033402, Sep 2020. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevResearch.2.033402>
- [15] S. H. Sack and M. Serbyn, “Quantum annealing initialization of the quantum approximate optimization algorithm,” 2021.
- [16] R. Shaydulin, I. Safro, and J. Larson, “Multistart methods for quantum approximate optimization,” *2019 IEEE High Performance Extreme Computing Conference (HPEC)*, Sep 2019. [Online]. Available: <http://dx.doi.org/10.1109/HPEC.2019.8916288>
- [17] F. G. S. L. Brandao, M. Broughton, E. Farhi, S. Gutmann, and H. Neven, “For fixed control parameters the quantum approximate optimization algorithm’s objective function value concentrates for typical instances,” 2018.
- [18] M. Alam, A. Ash-Saki, and S. Ghosh, “Accelerating quantum approximate optimization algorithm using machine learning,” in *Proceedings of the 23rd Conference on Design, Automation and Test in Europe*, ser. DATE ’20. San Jose, CA, USA: EDA Consortium, 2020, p. 686–689.
- [19] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, “Quantum computation by adiabatic evolution,” 2000.
- [20] G. Crooks, “Performance of the quantum approximate optimization algorithm on the maximum cut problem,” *arXiv: Quantum Physics*, 2018.
- [21] J. Larson and S. M. Wild, “Asynchronously parallel optimization solver for finding multiple minima,” *Mathematical Programming Computation*, vol. 10, no. 3, pp. 303–332, 2018.
- [22] —, “A batch, derivative-free algorithm for finding multiple local minima,” *Optimization and Engineering*, vol. 17, pp. 205–228, 2016.
- [23] M. Powell, “The bobyqa algorithm for bound constrained optimization without derivatives,” *Technical Report, Department of Applied Mathematics and Theoretical Physics*, 01 2009.
- [24] M. J. D. Powell, *A Direct Search Optimization Method That Models the Objective and Constraint Functions by Linear Interpolation*, 1994, pp. 51–67. [Online]. Available: <https://app.dimensions.ai/details/publication/pub.1046127469>
- [25] J. A. Nelder and R. Mead, “A Simplex Method for Function Minimization,” *The Computer Journal*, vol. 7, no. 4, pp. 308–313, 01 1965. [Online]. Available: <https://doi.org/10.1093/comjnl/7.4.308>
- [26] V. Akshay, D. Rabinovich, E. Campos, and J. Biamonte, “Parameter concentration in quantum approximate optimization,” 2021.