Vol.2021-QS-2 No.12
2021/3/29

A Study on the Leapfrogging Strategy and Parameters Fixing for the Quantum Approximate Optimization Algorithm on the Max-cut of n-regular Graph Instances

Xinwei Lee[1] and Dongsheng Cai[2]

[1,2] *University of Tsukuba, Tsukuba, Ibaraki.*

Abstract

The quantum approximate optimization algorithm (QAOA) has numerous promising applications on solving the combinatorial optimization problems on the near-term Noisy Intermediate Scalable Quantum (NISQ) devices. QAOA has a quantum-classical hybrid structure, with the quantum part consisting the parameterized alternating operator ansatz, and the classical part consist of an optimization algorithm optimizing the parameters to maximize the expectation value. This value depends highly on the parameters. This implies that a set of good parameters leads to an accurate solution of the given problem. However, at large circuit depth, it is difficult to achieve global optimization due to the multiple occurrence of local maxima. Therefore, we study the so-called leapfrogging strategy on solving the Max-cut problem for 3-regular graphs, which reuses the optimized parameters in larger graphs. Also, we propose a strategy of parameters fixing to increase the quality of the results as the circuit depth gets larger.

1 Introduction

Since the introduction of QAOA by Farhi et al. [1], it is widely known for its efficiency in solving combinatorial optimization problem on quantum computers. Among the problems, the Max-cut problem is heavily studied for its simple formulation and its deep relation to the Ising model. Since the Max-cut problem is an NP-complete problem, it is known that we are unable to solve it efficiently using classical computers, unless P=NP. Although QAOA does not give the exact solution to the Max-cut problem, the algorithm provides a heuristic approach for the problem. However, several hurdles continue to exist in obtaining a "good" solution for QAOA Max-cut. Solving large problems (graphs with large number of nodes) requires the deep circuit, but QAOA has difficulty at deep circuit due to the existence of many local maxima on the hypersurface of the expectation function.

In this paper, we use the leapfrogging strategy introduced [2] on regular graphs to tackle the problem, and verify our results using the Qiskit Aer QASM simulator. We also introduce a parameter fixing strategy to solve the problem of getting bad results at large circuit depths.

2 Solving Max-cut with QAOA

QAOA is inspired by the Quantum Adiabatic Algorithm [3], which focuses on evolving the initial Hamiltonian H_B to the problem Hamiltonian H_C, satisfying

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

where $s(t) \to 1$ as $t \to \infty$. The evolution in (1) is then discretized, which results in the idea of 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 (1):

$$|\psi_p(\gamma,\beta)\rangle = e^{-i\beta_p B}e^{-i\gamma_p C}\dots e^{-i\beta_1 B}e^{-i\gamma_1 C}|+\rangle^{\otimes n} \tag{2}$$

where $B = H_B$ and $C = H_C$. For the Max-cut problem of a graph $G = (V, E)$, they are given

ⓒ 2021 Information Processing Society of Japan

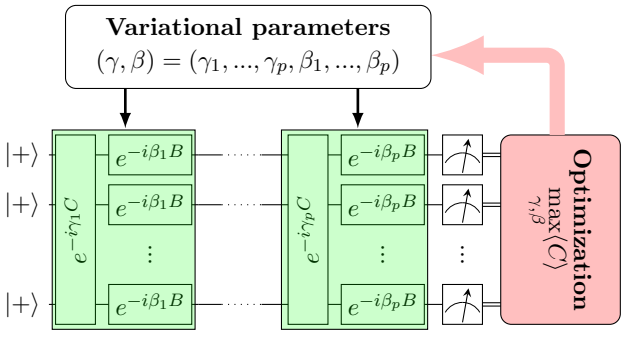

Figure 1: The quantum circuit and the schematic of the Quantum Approximate Optimization Algorithm (QAOA).

as

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

$$B = \sum_{j\in V} X_j \tag{4}$$

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

$$F_p(\gamma, \beta) = \langle \psi_p(\gamma, \beta) | C | \psi_p(\gamma, \beta) \rangle \tag{5}$$

Since (5) is parameterized by the angles γ and β, we can use a classical optimization algorithm to search for the angles which maximize F_p.

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

The approximation ratio α is defined as

$$\alpha = \frac{F_p(\gamma^*, \beta^*)}{C_{\text{true}}} \tag{7}$$

where C_{true} is the true Max-cut value for the graph to be solved. Figure 1 shows the schematic of QAOA.

The positive integer p is known as the circuit depth, and it is shown theoretically that as $p \to \infty$, the maximum of F_p will approach C_{true}. However, for example, previous report [4] states that at large p, it is difficult to maximize F_p due to: 1. The point to be optimized is more likely to be trapped inside a local maxima; 2. The optimizer takes longer as it has more parameters to consider. These drawbacks discourage us to increase the circuit depth.

3 The leapfrogging strategy for regular graphs

Farhi et al. [1] observe that at $p = 1$, for regular graphs, for each local term in (5), the operators which are not involved in the edge (j, k) will be canceled out and does not contribute to F_1. This further simplifies F_1 to the linear combination of the expectation value of the 3 types of subgraphs in 3-regular graphs.

$$\begin{aligned} F_1(\gamma_1, \beta_1) =& w_{\text{I}} F_{\text{I}}(\gamma_1, \beta_1) + w_{\text{II}} F_{\text{II}}(\gamma_1, \beta_1) \\ &+ w_{\text{III}} F_{\text{III}}(\gamma_1, \beta_1) \end{aligned} \tag{8}$$

I, II and III represent the 3 types of subgraphs, w is their respective total weights and F is their local expectations. At $p = 1$, the local expectation functions F_{I}, F_{II} and F_{III} will be exactly the same for all 3-regular graphs, as they only depend on the structure of the 3 types of subgraphs. Based on this fact, Brandao et al. [2] then propose a leapfrogging strategy to inherit optimized angles from typical graph instances with smaller number of nodes to that with larger number of nodes. Typical instance means that the instance has large number of nodes and high proportion of the type III subgraphs (large w_{III}) compared to I and II, hence the overall expectation function F_1 concentrates towards F_{III}. Since for typical instances, their F_1 is essentially the same regardless of the number of nodes, they can use the same angles to produce the same expectation. From $p = 2$ onwards, the subgraphs considered become more complex, but the leapfrogging strategy will still be applicable if the proportion for the tree-like subgraph is high.

We solve the Max-cut on unweighted 3-regular instances with 6, 8, 12 and 16 nodes using QAOA with the Nelder-Mead optimizer. We choose the graphs such that they have higher proportion of the type III subgraph. We then compare the approximation ratio α between random initial angles and the leapfrogging method. For the random initialization method, we use 20 random angles as the starting point to the QAOA. The results show that α span a range of different values when different initial angles are used in QAOA. Therefore, QAOA does not guarantee high α when solved using

ⓒ 2021 Information Processing Society of Japan

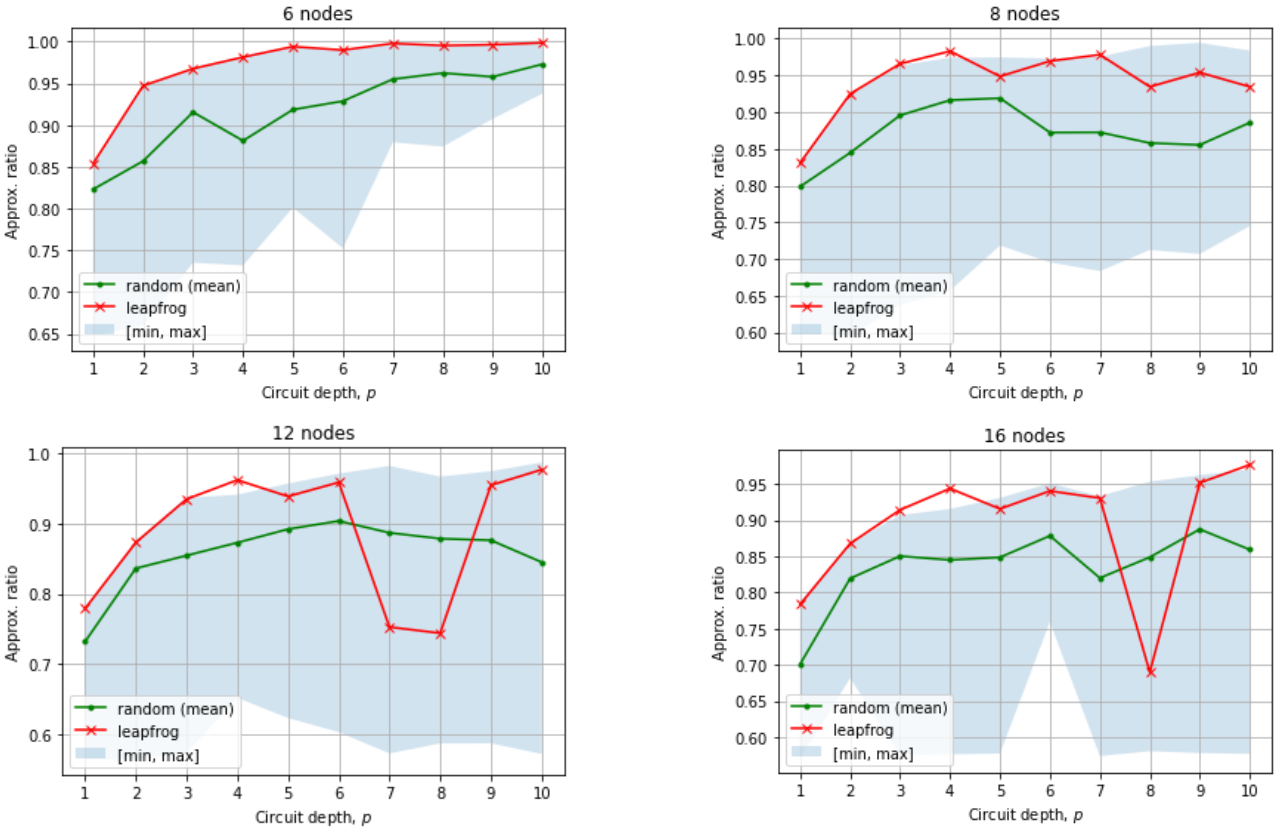

Figure 2: Reults of the QAOA on 6, 8, 12 and 16 nodes instances with the leapfrogging strategy applied. The shaded region is the range of data spanned by the random initialization method.

random initial angles. For the leapfrogging method, we first use the random method on solving the 6-node graph. Then, we pass the angles that output the highest α in the 6-node graph as the initial angles to the 8-node graph. We optimize those sets of angles and pass them to the 12-node graph, and then repeat the same for 16.

Figure 2 shows the results for the QAOA simulation. For most of the circuit depths, the value of α for the leapfrogging method is higher than the α averaged out of 20 times for the random initialization method. At $p = 1$, the results agree with [1] which states that $\alpha \geq 0.6924$ for 3-regular graphs. For the instances with inherited angles (8, 12 and 16 nodes), as the depth increases, the approximation ratio increases until around $p = 4$ with $\alpha \approx 0.95$ before the increasing trend stops. For the 12-node and 16-node graphs, α drops dramatically to $\alpha \approx 0.7$ at $p = 7$ and $p = 8$ ($p = 8$ for 16-node) when the inherited angles are bad. For the random initialization method, the maximum α among the 20 different random angles exhibits the increasing trend as the circuit depth increases.

As discussed in [2], the leapfrogging method is only suitable for "typical instances". This causes the leapfrogging method to sometimes fail at larger circuit depth for graphs with small number of nodes, as the subgraph considered gets more and more different as the depth increases. Despite that, the advantage of the leapfrogging method holds at small depth as it does not require trying multiple different angles each time a graph instance is solved to obtain the global maxima.

4 Parameters fixing

Knowing the fact that as the depth increases, the subgraphs considered gets more different, it is required to select better angles when applying QAOA with larger circuit depth.

ⓒ 2021 Information Processing Society of Japan

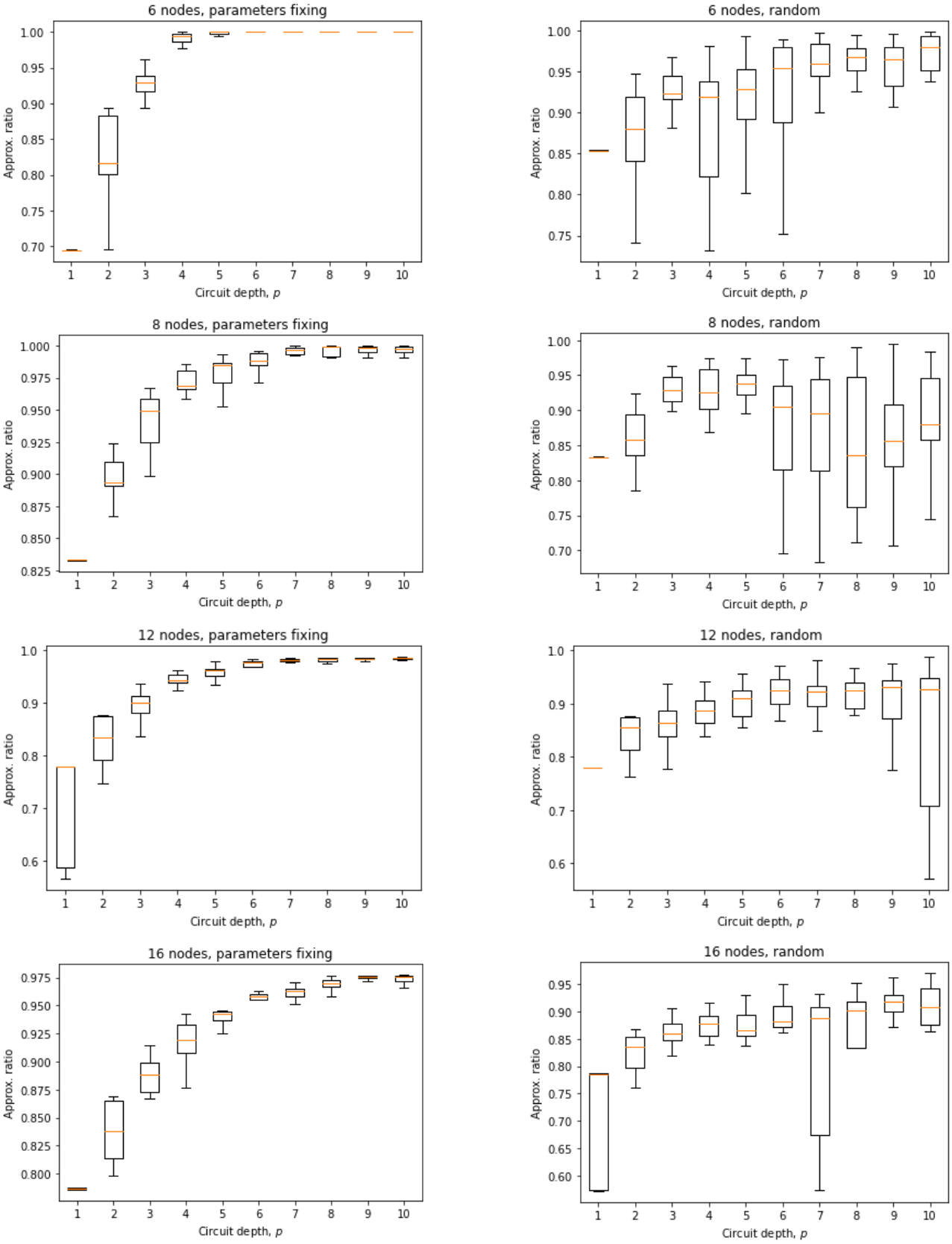

Figure 3: Boxplots of the results of QAOA with the parameters fixing strategy applied.

ⓒ 2021 Information Processing Society of Japan

Several works [5][6][7] have shown the relationship between the optimal angles and the circuit depth of QAOA which resembles the linear annealing scheme. We then attempt to extrapolate the angles of depth $p+1$ with the optimal angles obtained from depth p. We first find the optimal angles at $p = 1$, then we fix this set of angles and add another randomized (γ, β) pair, and pass this set of angles as initial angles to $p = 2$. Then repeat for larger p.

Same instances from the previous section are solved using QAOA with the parameters fixing method applied. 20 different randomized (γ, β) pairs were added to the optimal angles and the one which outputs the highest α is chosen and is passed to the next depth along with the previous optimal angles. Figure 3 shows the result of the QAOA simulation with the parameters fixing method. The result is compared with the one obtained using the random initialization method. For the random initialization method, α does not necessarily increase in overall when the circuit depth increases. The deviation of α is also not consistent. We find out that by fixing the optimal angles at smaller depth, the value of α increases steadily as the circuit depth increases. It is also worth noting that at larger p, the deviation of α is relatively small, compared to those at smaller p. This implies that if the angles up until depth p are optimal, then the significance of the angles at depth $p+1$ will be smaller, as different angles at $p+1$ will still produce the similar value of α.

Figure 4 shows how the landscape of the expectation function $F_p(\gamma, \beta)$ for the 6-node graph changes when the parameters fixing method is applied. The landscapes are plotted starting from $p = 1$, with the angles (γ_1, β_1) varied. The angles with highest expectation are then taken as the optimal angles and are fixed in plotting the landscape of the following depths, with the angles (γ_2, β_2) varied. It is then repeated until $p = 4$. The landscape at depth p has (γ_p, β_p) as the independent variables with all the previous optimal angles as constants. We observe that as the depth increases while the optimal angles are fixed, the maximum points in the landscape appear to merge together. Finally, at $p = 4$ the landscape becomes river-like and the maximum points become a maximum line in the landscape. The maximum value in the landscape also increases as the depth increases (Table 1). We also plotted the landscape for the 8-, 12- and 16-node graphs and they exhibit the same pattern shown in Figure 4.

Table 1: Maximum values of the expectation landscape at different circuit depths.

p	**Max. expectation**
1	5.939
2	6.453
3	6.612
4	6.613

The landscapes in Figure 4 are exactly the ones which are searched by the optimizers to maximize $F_p(\gamma, \beta)$ when the parameters fixing method is applied. The parameters fixing method consequently result in a search on a changed landscape instead of the original landscape (the one with all the angles varied). With the landscape evolving to a river-like surface, the maximum points into a maximum line, it will be easier for the optimizers to search for the maximum value and less chance to be trapped inside a local maximum. This explains the better-quality results of the parameters fixing method compared to the random initialization method. However, the parameters fixing method has the disadvantage of high computational cost due to the need of optimizing 20 different sets of angles and choose the best one for each level of depth. The computational cost is about the same for the random initialization method, as it also requires optimizing multiple sets of different angles.

5 Conclusion

We have applied the leapfrogging strategy to 3-regular graphs and the result shows it is able to approximate better than the random approach at smaller circuit depth for small graphs. However, for small graphs at large circuit depth, the leapfrogging strategy does not work well as it is designed to work on large typical instances. The subgraphs involved in

ⓒ 2021 Information Processing Society of Japan

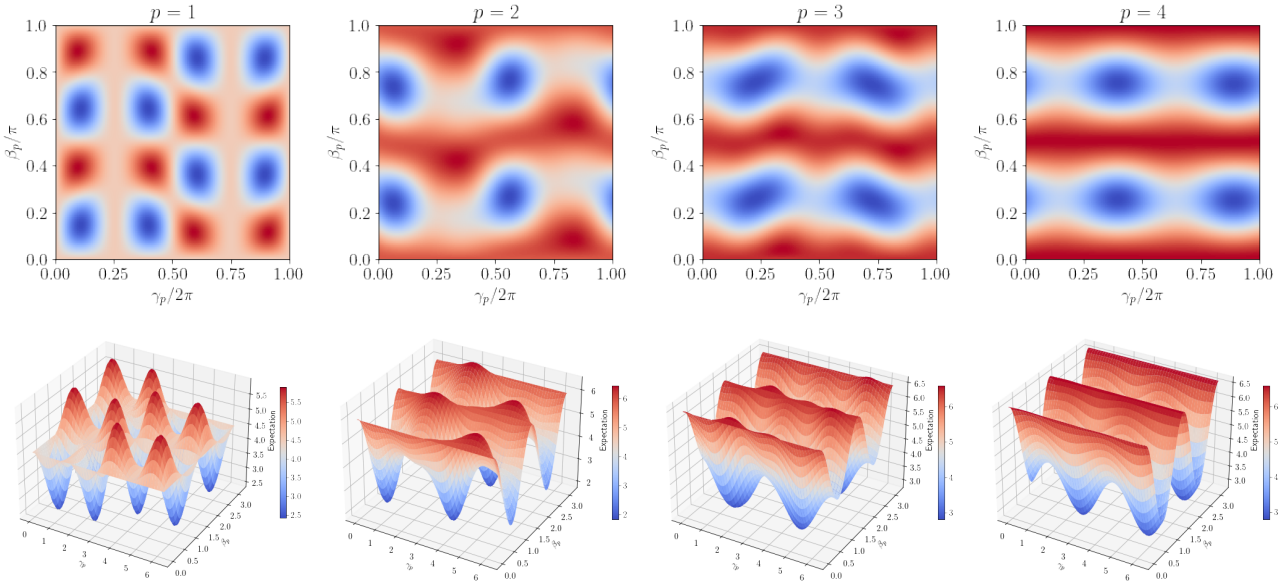

Figure 4: Heatmap and landscape of the expectation function of a 6-node graph as the circuit depth increases, with the parameters fixing strategy applied.

the expectation function gets different at larger circuit depth for small graphs.

On the other hand, we show the results of the proposed parameters fixing strategy. The strategy improves the results at large circuit depth by altering the landscape searched by the optimizers. We observe that using this strategy, the landscape of the expectation function evolves into a river-like structure as the circuit depth increases. This consequently eliminates the local maxima at larger circuit depths and improves the search of the optimizer.

In the future, we will study the effect of combining the two strategies as the nodes and the depths of the QAOA increase.

References

[1] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm, 2014, arXiv:1411.4028.

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

[3] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Quantum computation by adiabatic evolution. 2000, arXiv:quant-ph/0001106.

[4] G. G. Guerreschi and A. Y. Matsuura. Qaoa for max-cut requires hundreds of qubits for quantum speed-up. *Scientific Reports*, 9(1), May 2019.

[5] G. Crooks. Performance of the quantum approximate optimization algorithm on the maximum cut problem. *arXiv: Quantum Physics*, 2018.

[6] Leo Zhou, Sheng-Tao Wang, Soonwon Choi, Hannes Pichler, and Mikhail D. Lukin. Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices. *Phys. Rev. X*, 10:021067, Jun 2020.

[7] J. Cook, S. Eidenbenz, and Andreas Bärtschi. The quantum alternating operator ansatz on maximum k-vertex cover. *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pages 83–92, 2020.

ⓒ 2021 Information Processing Society of Japan