A note on quantum approximate optimization algorithm

Zhengjun Cao

Abstract. The general quantum approximate optimization algorithm (QAOA) produces approximate solutions for combinatorial optimization problems. The algorithm depends on a positive integer p and the quality of approximation improves as p is increased. In this note, we put some questions about the general QAOA. We also find the recursive QAOA for MaxCut problem is flawed because all quantum gates involved in the algorithm are single qubit gates. No any entangling gate is used, which results in that the quantum computing power cannot be certified for the problem.

Keywords: Combinatorial optimization, QAOA, RQAOA, MaxCut, Computational basis, Uniform superposition

1 Introduction

Quantum computer is regarded as the biggest threat to modern public key cryptography, due to Shor algorithms for prime factorization and discrete logarithms [1]. Thirty years later, however, the hypothetical device remains far away from becoming reality. We are now facing the embarrassing situation. On the one hand, there were many announcements of success in manufacturing quantum computers, including IBM 133 qubits on the Heron chip and Google 70 operational qubits prototype. Both have stepped across the threshold of quantum supremacy (53 qubits). On the other hand, there is no guarantee of success in running these devices to solve an actual numerical computation problem. There must be some deep reasons for this situation. The misunderstandings about some quantum algorithms (Shor's algorithm, Grover's algorithm [2], etc.), could be the main reason for the conflict between ideal and reality. Ten years ago, we investigated the Shor's algorithm and put some questions about its variation. Unfortunately, the investigation has attracted little attention. In this note, we further put some questions about the general quantum approximate optimization algorithm (QAOA) and the recursive QAOA.

The general quantum approximate optimization algorithm (QAOA), introduced by Farhi, Goldstone and Gutmann [4, 5], was designed to solve hard combinatorial optimization problems. However, it has been known that QAOA has limited performance for several instances [6–10]. The recursive QAOA (RQAOA) proposed by Bravyi et al. [11], aims to overcome the limitations of QAOA. In 2022, Patel et al. [12] discussed the reinforcement learning assisted RQAOA. Very recently, Bae and Lee [13] have tried to analytically prove that the RQAOA is more competitive than the QAOA to solve the MaxCut problem for complete graphs. In this note, we remark that there are some questions about the general QAOA should be claimed. We also find the RQAOA for MaxCut problem is flawed because all quantum gates involved in the algorithm are single qubit gates. No any entangling gate

Department of Mathematics, Shanghai University, Shanghai, 200444, China. Email: caozhj@shu.edu.cn

is used. In this case, one cannot certify the quantum computing power of RQAOA for the MaxCut problem.

2 Preliminaries

2.1 Single-qubit states and operators

The state of a single qubit can be described by a two-dimensional column vector of unit norm. Thus $\begin{bmatrix} \alpha \\ \beta \end{bmatrix}$ represents a qubit state if α and β are complex numbers satisfying $|\alpha|^2 + |\beta|^2 = 1$. The basic two quantum states are taken to correspond to the two states of a classical bit, i.e., bit 0 $\leftrightarrow |0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, bit 1 $\leftrightarrow |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$. The basic single-qubit operations include

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad T = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}, \quad S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} = T^2,$$

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = HT^4H, \quad Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} = T^2HT^4HT^6, \quad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = T^4,$$

where X, Y, Z are called Pauli operators, and H is called Hadamard gate. Now, we have

$$H|0\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \overset{\text{def}}{=} |+\rangle.$$

$$H|1\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 0 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ -1 \end{bmatrix} = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \overset{\text{def}}{=} |-\rangle.$$

Any unitary matrix can be written as a sequence of three rotations:

$$\begin{aligned} R_z(\theta) &= e^{-i\theta Z/2} = \begin{bmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{bmatrix}, \\ R_x(\theta) &= e^{-i\theta X/2} = \begin{bmatrix} \cos(\theta/2) & -i\sin(\theta/2) \\ -i\sin(\theta/2) & \cos(\theta/2) \end{bmatrix}, \\ R_y(\theta) &= e^{-i\theta Y/2} = \begin{bmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{bmatrix}. \end{aligned}$$

2.2 Two-qubit states and operators

Given two separate qubits $\psi = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$, $\phi = \begin{bmatrix} \gamma \\ \delta \end{bmatrix}$, the corresponding two-qubit state is given by the tensor product of vectors, which is defined as follows

$$\psi \otimes \phi = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \otimes \begin{bmatrix} \gamma \\ \delta \end{bmatrix} = \begin{bmatrix} \alpha \begin{bmatrix} \gamma \\ \delta \end{bmatrix} \\ \beta \begin{bmatrix} \gamma \\ \delta \end{bmatrix} \end{bmatrix} = \begin{bmatrix} \alpha\gamma \\ \alpha\delta \\ \beta\gamma \\ \beta\delta \end{bmatrix}.$$

The basis for two-qubit states is formed by the tensor products of one-qubit basis states, i.e.,

$$\text{string } 00 \leftrightarrow |00\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \text{string } 01 \leftrightarrow |01\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix},$$

$$\text{string } 10 \leftrightarrow |10\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \quad \text{string } 11 \leftrightarrow |11\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}.$$

As in the single-qubit case, any unitary transformation is a valid operation on qubits. A unitary transformation on n qubits is a matrix $U$of size $2^n \times 2^n$, such that $U^{-1} = U^\dagger$. The CNOT (controlled-NOT) gate is a commonly used two-qubit gate as well as the SWAP gate.

$$\text{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \text{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Sometimes, two-qubit gates can be described by the tensor product of some known single-qubit gates. For example, $H \otimes H, H \otimes \mathbf{1}, H \otimes Z$. Not all two-qubit gates can be written as the tensor product of single-qubit gates. Such a gate is called an *entangling gate*, for example, the CNOT gate. A controlled-not gate can be generalized to arbitrary gates, which acts as identity unless a specific qubit is 1. The gates H, T and CNOT form a universal gate set on many qubits because any general unitary transformation can be broken into a series of two qubit rotations.

2.3 Equal-superposition

The term superposition simply refers to a linear combination. Note that

$$H|0\rangle \otimes H|0\rangle = \left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right) \otimes \left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right) = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle).$$

Let $H^{\otimes n} = \underbrace{H \otimes H \otimes \cdots \otimes H}_{n}$. The n-qubit equal-superposition for the standard basis can be generated as follows

$$H^{\otimes n}|00\cdots 0\rangle = \frac{1}{\sqrt{2^n}}\left(|00\cdots 0\rangle + |00\cdots 1\rangle + \cdots + |11\cdots 1\rangle\right)$$

An equal-superposition state is often the starting state for many quantum algorithms.

2.4 Unitary operations

The only way to change qubits without measuring is to apply a unitary operation. Quantum computations can be created by designing unitary operations in sequence, each of which is composed of smaller operations. Precisely explaining quantum speed-up remains an unresolved issue. Some results show that entanglement must be present when speed-up occurs.

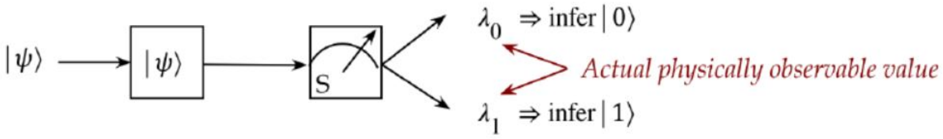
2.5 Projective measurements

Given the state $|0\rangle$ and the computational basis $|+\rangle, |-\rangle$, we have $|0\rangle = \frac{1}{\sqrt{2}}|+\rangle + \frac{1}{\sqrt{2}}|-\rangle$. Applying the projective measurement to the state, the probability of seeing $|+\rangle$ is just $|\frac{1}{\sqrt{2}}|^2 = \frac{1}{2}$. So does that of seeing $|-\rangle$. To examine a qubit's state, one must measure. A measurement outcome is random, and is one of the measurement-basis vectors. Measurement changes the qubit's state into one of the basis vectors.

Consider the two-qubit state $|\psi\rangle = \frac{1}{\sqrt{2}}|01\rangle + \frac{1}{\sqrt{2}}|10\rangle$. Suppose we use the two-qubit standard basis $|00\rangle, |01\rangle, |10\rangle, |11\rangle$ for two-qubit measurement. The outcomes of measurement are the measurement-basis vectors: $|00\rangle, |01\rangle, |10\rangle, |11\rangle$. The state $|01\rangle$ occurs with probability $|\frac{1}{\sqrt{2}}|^2$. So does the state $|10\rangle$. Physically implementing a 2-qubit measurement is difficult, but a 2-qubit standard-basis measurement can be done in two stages of standard-basis 1-qubit measurements.

2.6 Numerical interpretations

In real physical devices, the associated eigenvalue (a real number) is a physical quantity that's observable, like energy or frequency. Every unique outcome (a vector) has a unique accompanying physical value. But in quantum computing, such eigenvalues play no role. We only care about the resulting eigenvectors. For example, a qubit state is measured and interpreted as below.


$|\psi\rangle$
$|\psi\rangle$
S
λ_0 ⇒ infer $|0\rangle$
Actual physically observable value
λ_1 ⇒ infer $|1\rangle$

Note that, the state $|0\rangle$ is conventionally interpreted as the bit 0, and $|1\rangle$ corresponds to bit 1. The measurement outcome of an n-qubit state is numerically interpreted as an n-bit string, which is generally interpreted as an integer, not a decimal.

3 The general quantum approximate optimization algorithm

A combinatorial optimization problem can be interpreted as n bits and m clauses. The number of satisfied clauses for string z is defined by the objective function,

$$C(z) = \sum_{\alpha=1}^{m} C_\alpha(z) \tag{1}$$

where $z = z_1 z_2 \cdots z_n$ is the bit string and $C_\alpha(z) = 1$ if z satisfies clause α and 0 otherwise. Satisfiability asks if there is a string that satisfies every clause. MaxSat asks for a string that maximizes the objective function. Approximate optimization asks for a string z for which $C(z)$ is close to the maximum of C.

More concretely, for problems defined on graphs the cost function can be written as $C = \sum_{<i,j>} C_{ij}$, where C_{ij} is the cost function associated with the edge connecting vertices i and j. If the bit values on the vertices are such that the edge constraint is satisfied then $C_{ij} = 1$, if it is not then $C_{ij} = 0$. The goal is to make C big. For Max-Cut the associated cost function is $C_{ij}^{MC} = b_i + b_j - 2b_i b_j$, where each b_i is 0 or 1. This cost is satisfied when the bits disagree. For Maximum Independent Set on d-regular graphs, the choice of cost function is $C_{ij}^{MIS} = \frac{1}{2d}(b_i + b_j) - b_i b_j$.

In 2014, Farhi, Goldstone, and Gutmann designed a quantum algorithm for approximate optimization (QAOA). They also studied the algorithm's performance in special cases of MaxCut and proposed an alternate form of the algorithm geared toward finding a large independent set of vertices of a graph. In the QAOA, the initial state is set as $|s\rangle = |+\rangle^{\otimes n} = \frac{1}{\sqrt{2}} \sum_z |z\rangle$. Define a unitary operator $U(C, \gamma)$ which depends on an angle γ,

$$U(C, \gamma) = e^{-i\gamma C} = \prod_{\alpha=1}^{m} e^{-i\gamma C_\alpha} \tag{2}$$

Define the operator B which is the sum of all single bit σ^x operators,

$$B = \sum_{j=1}^{n} \sigma_j^x \tag{3}$$

where $\sigma_j^x = X_j$, the Pauli X operator acting on qubit j. Then define the β dependent product of commuting one bit operators

$$U(B, \beta) = e^{-i\beta B} = \prod_{j=1}^{n} e^{-i\beta \sigma_j^x} \tag{4}$$

Pick $2p$ angles $\beta_1, \cdots, \beta_p, \gamma_1, \cdots, \gamma_p$ to modulate the state

$$|\boldsymbol{\gamma}, \boldsymbol{\beta}\rangle = U(B, \beta_p) U(C, \gamma_p) \cdots U(B, \beta_1) U(C, \gamma_1) |s\rangle \tag{5}$$

The associated QAOA objective function is $\langle \boldsymbol{\gamma}, \boldsymbol{\beta} | C | \boldsymbol{\gamma}, \boldsymbol{\beta} \rangle$, and the goal is to find $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ to make this big. Let the expectation of C in this state be

$$F_p(\boldsymbol{\gamma}, \boldsymbol{\beta}) = \langle \boldsymbol{\gamma}, \boldsymbol{\beta} | C | \boldsymbol{\gamma}, \boldsymbol{\beta} \rangle \tag{6}$$

and

$$M_p = \max_{\boldsymbol{\gamma}, \boldsymbol{\beta}} F_p(\boldsymbol{\gamma}, \boldsymbol{\beta}) \tag{7}$$

Finally, it argues that

$$M_p \geq M_{p-1}, \quad \lim_{p \to \infty} M_p = \max_z C(z)$$

4 Questions about the general QAOA

There are four concerns about the general QAOA.

- The first concern is about the method to construct the involved transformations. In the expression

  $$U(B, \beta_1) U(C, \gamma_1) |s\rangle = \prod_{j=1}^{n} e^{-i\beta_1 \sigma_j^x} \prod_{\alpha=1}^{m} e^{-i\gamma_1 C_\alpha} |s\rangle$$

  the value C_α is originally defined by the correlation of classical string z and the α-th clause.

That is, given string z and the α-th clause, $C_\alpha(z) = 1$ or 0. In the quantum scenario, the superposition $|s\rangle$ cannot be mathematically interpreted as an n bits string before quantum measurement. So, how to determine the values $C_1, \cdots, C_m$ so as to construct the transformation $U(C, \gamma_1)$? Note that *a quantum state could be ambiguous but a transformation should be explicit.*

- The second concern is about the method to construct the quantum oracle C to implement the classical function $C(z)$ so as to modulate the wanted state $C|\boldsymbol{\gamma}, \boldsymbol{\beta}\rangle$. So far, such a problem is only hypothetically claimed. The details have not been described or even interpreted.
- The third concern is about the method to measure the state $C|\boldsymbol{\gamma}, \boldsymbol{\beta}\rangle$, which is a superposition. Namely, how to choose the projective measurements so as to determine the expectation $\langle\boldsymbol{\gamma}, \boldsymbol{\beta}|C|\boldsymbol{\gamma}, \boldsymbol{\beta}\rangle$? Or how to determine what projective measurements corresponding to the expectation?
- The fourth concern is about the method to numerically interpret the measured values. If the observed energy value λ cannot be definitely inferred as $|0\rangle$ or $|1\rangle$, which interpretation is more preferred?

5 Recursive QAOA for MAX-CUT

Let $G = (V, E)$ be a graph with the set of vertices $V = \{1, 2, \cdots, n\}$ and the set of edges $E = \{(i, j) : i, j \in V\}$. The MaxCut problem aims to split V into two disjoint subsets such that the number of edges spanning the two subsets is maximized. The problem can be formulated by maximizing the cost function

$$C(x) = \frac{1}{2} \sum_{(i,j)\in E} (1 - x_i x_j)$$

for $x = (x_1, x_2, \cdots, x_n) \in \{-1, 1\}^n$. This cost function can be further converted to a quantum version

$$H_C = \frac{1}{2} \sum_{(i,j)\in E} (I - Z_i Z_j)$$

where Z_i is the Pauli operator Z acting on the i-th qubit.

5.1 Description of the RQAOA

In 2023, Bae and Lee [13] tried to prove that the RQAOA is more competitive than the QAOA to solve the MaxCut problem for complete graphs. They rephrased the two algorithms as follows.

Algorithm 1 (QAOA$_p$).

1. Initialize the quantum processor in $|+\rangle^{\otimes n}$.
2. Generate the state

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

where $H_B = \sum_{i=1}^n X_i$, X_i is the Pauli operator X (bit-flip) acting on the i-th qubit.

3. Compute the expectation value $F_p(\beta, \gamma) = \langle \psi_p(\beta, \gamma) | H_C | \psi_p(\beta, \gamma) \rangle$.
4. Find $(\beta^*, \gamma^*) = \mathrm{argmax}_{(\beta,\gamma)} F_p(\beta, \gamma)$ using a classical optimization algorithm.

The approximation ratio $r = \frac{F_p(\beta^*,\gamma^*)}{C_{\max}}$, where $C_{\max} = \max_{x \in \{-1,1\}^n} C(x)$. Let

$$H_n = \sum_{(i,j)\in E} J_{i,j} Z_i Z_j$$

which is defined on a graph $G_n = (V, E)$ with $|V| = n$, where $J_{i,j} \in \mathbb{R}$ are arbitrary.

Algorithm 2 (RQAOA$_p$).

1. Apply the QAOA$_p$ to find the state $|\psi_p(\beta^*, \gamma^*)\rangle$ which maximizes $\langle \psi_p(\beta, \gamma) | H_n | \psi_p(\beta, \gamma) \rangle$.
2. Compute $M_{i,j} = \langle \psi_p(\beta^*, \gamma^*) | Z_i Z_j | \psi_p(\beta^*, \gamma^*) \rangle$ for every edges $(i, j) \in E$.
3. Choose a pair (k, l) which maximizes the magnitude of $M_{i,j}$.
4. Call the QAOA$_p$ recursively to maximize the expected value of

$$H_{n-1} = \sum_{(i,l)\in E'_0} J'_{i,j} Z_i Z_l + \sum_{i,j \in E'_1} J'_{i,j} Z_i Z_j$$

where $E'_0 = \{(i, l) : (i, k) \in E\}$, $E'_1 = \{(i, j) : i, j \neq k\}$, and

$$J'_{i,j} = \mathrm{sgn}(M_{k,l}) J_{i,k} \text{ if } (i, l) \in E'_0, \quad J'_{i,j} = J_{i,j} \text{ if } (i, j) \in E'_1.$$

5. The recursion stops when the number of variables reaches some suitable threshold value $n_c \ll n$. Find

$$\boldsymbol{X}^* = \mathrm{argmax}_{\boldsymbol{X} \in \{-1,1\}^{n_c}} \langle \boldsymbol{X} | H_{n_c} | \boldsymbol{X} \rangle$$

by a classical algorithm.

6. Reconstruct the approximate solution $\tilde{\boldsymbol{X}} \in \{-1, 1\}^n$ from $\boldsymbol{X}^*$.

5.2 Questions about RQAOA

Though the RQAOA is interesting, we find it is not unquestionable.

- It's easy to find that all quantum gates (Hadamard gate, bit-flip gate, phase-flip gate) involved in the algorithm are single-qubit gates without any entangling gate. How to certify the quantum computing power in the algorithm?

  "If all you had in a quantum computer were single-qubit gates, then a calculator and certainly a classical supercomputer would dwarf its computational power" — Azure Quantum Documentation.

- Note that the coefficients $J_{i,j} \in \mathbb{R}$. How to numerically interpret the final measurement outcome if these coefficients are not integers? So far, we have not found any explicit specification to interpret a multiple qubits state as a decimal.

6 Conclusion

In this note, we put some questions about the general quantum approximate optimization algorithm and the recursive quantum approximate optimization algorithm for Max-Cut problem. The findings could be helpful for the future works on designing such algorithms.

References

[1] P. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM J. Comput., 26(5), 1484-1509, 1997.

[2] L. Grover. A fast quantum mechanical algorithm for database search. Proceedings 28th Annual Symposium on the Theory of Computing (STOC) 1996, pp. 212-219.

[3] Z. J. Cao and Z. F. Cao. On Shor's factoring algorithm with more registers and the problem to certify quantum computers. IACR Cryptol. ePrint Arch. 2014, 721.

[4] E. Farhi, J. Goldstone, and S. Gutmann. A quantum approximate optimization algorithm. arXiv:1411.4028, 2014.

[5] E. Farhi, J. Goldstone, and S. Gutmann. A quantum approximate optimization algorithm applied to a bounded occurrence constraint problem. arXiv:1412.6062, 2015.

[6] M. Hastings. Classical and quantum bounded depth approximation algorithms. arXiv:1905.07047, 2019.

[7] E. Farhi, D. Gamarnik, and S. Gutmann. The quantum approximate optimization algorithm needs to see the whole graph: A typical case. arXiv:2004.09002, 2020.

[8] E. Farhi, D. Gamarnik, and S. Gutmann. The quantum approximate optimization algorithm needs to see the whole graph: worst case examples. arXiv:2005.08747, 2020.

[9] K. Marwaha, Local classical max-cut algorithm outperforms $p = 2$ qaoa on high-girth regular graphs, Quantum 5, 437, 2021.

[10] B. Barak and K. Marwaha. Classical algorithms and quantum limitations for maximum cut on high-girth graphs. arXiv:2106.05900, 2021.

[11] S. Bravyi, A. Kliesch, R. Konig, and E. Tang. Obstacles to state preparation and variational optimization from symmetry protection, Physical Review Letters 125, 260504, 2020.

[12] Y. Patel, S Jerbi, T. Back, and V. Dunjko. Reinforcement Learning Assisted Recursive QAOA. arXiv:2207.06294, 2022.

[13] E. Bae and S. Lee. Recursive QAOA outperforms the original QAOA for the MAX-CUT problem on complete graphs. arXiv:2211.15832, 2023.