A Tutorial on Quantum Approximate Optimization Algorithm (QAOA): Fundamentals and Applications

Jaeho Choi
School of Computer Science and Engineering
Chung-Ang University
Seoul, Republic of Korea
jaehochoi2019@gmail.com

Joongheon Kim
School of Electrical Engineering
Korea University
Seoul, Republic of Korea
joongheon@korea.ac.kr

***Abstract*—Over the past few years, many researchers around the world have been keen to know the potential and efficiency of quantum computers. The researchers have focused on specific issues that classical computers cannot solve or issues that quantum computers can handle in a better way. Among these various attractive research topics in quantum computers, this paper introduces the Quantum Approximate Optimization Algorithm (QAOA) which guarantees relatively considerable performances in many combinatorial optimization problems. For the comprehensive understanding of QAOA, this paper also describes the approximate optimization, the Quantum Alternating Operator Ansatz, and applications. Besides the theories of QAOA and Quantum Alternating Operator Ansatz, this paper explains the applications of QAOA to major combinatorial optimization problems such as maximum cut (MaxCut) problem and the max-independent set (MIS) problem.**

I. Introduction

The needs and interests in quantum computers that can transcend the power of supercomputers have been long around. To meet these needs and interests, researchers have conducted the research in various research directions. Based on the efforts by large-scale IT companies, the commercialization of quantum computers can be realized, near future. The interests in quantum algorithms are also increasing with this trend. Among various quantum computing research topics, we introduce the fundamentals and applications of the Quantum Approximate Optimization Algorithm (QAOA), which has been recently studied in quantum computing research societies.

QAOA was first introduced to the world in 2014 by *Farhi et al.* [1]. QAOA begins with mapping the objective function to Hamiltonian to bring the problem into Hilbert space. We can obtain the expectation value of Hamiltonian using quantum mechanical techniques in the Hilbert space. Then, we can reach the solution via iteration processes which obtain the parameters that optimize the expectation value of Hamiltonian. QAOA is a quantum gate model algorithm, ensuring (i) a simple and monotonous structure and (ii) relatively good performance.

Since the appearance of QAOA in the world, various topics related to QAOA have been proposed. The summary of related results about QAOA are as follows, i.e., the topic of parameter setting by *Yang et al.* [2], the topic of compilation by *Venturelli et al.* [3] and *Oddi et al.* [4], the topic of combinatorial optimization by *Hadfield et al.* [5], the topic of factoring by *Anschuetz et al.* [6], and the topic of deep learning by *Verdon et al.* [7]. Based on the research results in these related work, we are interested in algorithmic understanding of QAOA and its applications. In terms of applications to QAOA, we discuss about QAOA-based solution approaches to the maximum cut (MaxCut) problem and the max-independent set (MIS) problem.

II. Approximate Optimization

This section presents why approximate optimization is required. Moreover, its related backgrounds are also explained. Lastly, we also explain why quantum mechanical approaches are needed in the approximate optimization.

In combinatorial optimization problems, we try to maximize or minimize the objective function [8]. However, there are computational difficulties and limitations to obtain the optimization solutions in classical ways in some specific problems. For the cases, we obtain approximate solutions using heuristics (e.g., genetic algorithms, tabu search, and simulated annealing) or constant-gap approximation algorithms. In particular, NP-optimization (NPO) problems are well-known representative examples of these problems.

In general, an approximation algorithm can find an approximate solution in polynomial-time computation. In addition, the performance of an approximation algorithm can be presented by an approximation ratio, thus, it is called constant-gap approximation. An approximation algorithm which guarantees fantastic performance has an approximation ratio close to 1 [9]. For example, the approximation ratio (represented by R) for the maximization problem can be denoted as follows:

$$\frac{f(x)}{\max_x f(x)} \geq R, \tag{1}$$

where $f(x)$ is an objective function and $\max_x f(x)$ is the optimal value with optimal solution x_{op}. Notice that this approximation ratio R in the maximization problem is greater than or equal to 0 and less than or equal to 1, i.e., $0 \leq R \leq 1$. However, there is a limit to the improvement of the approximation ratio R. This is widely known as the hardness of approximation [10]. For example, the limit of the approximation ratio for the MaxCut problem is $16/17$ (≈ 0.94117), as well

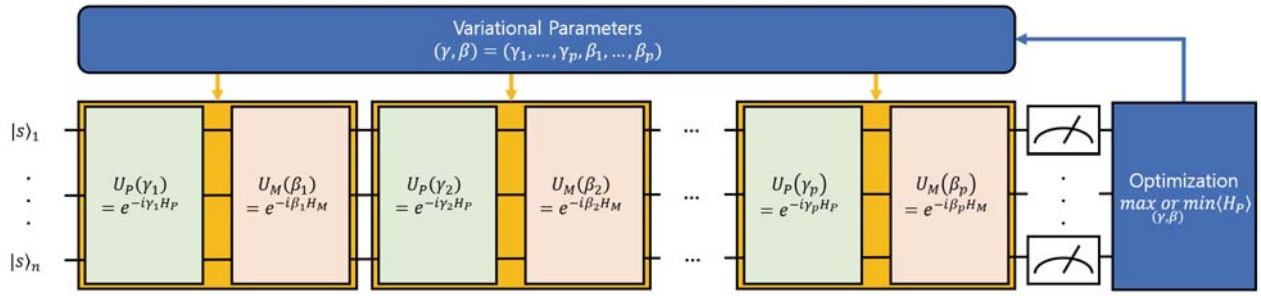

Fig. 1: p-level Quantum Approximate Optimization Algorithm (QAOA$_p$) [1].

studied in [11]. As the other example, the current best record is the Goemans-Williamson algorithm, which guarantees an approximation ratio of about 0.87856 [12].

Many researchers have attempted to reach the limit of the approximation ratio and have made new approaches. The most well known approaches are to utilize *quantum annealing algorithm* and *quantum gate model algorithms*, inspired by quantum mechanical approaches. The quantum gate model algorithm has a performance guarantee and has some advantages in terms of implementation, comparing to heuristic-based methods. In this paper, we study about the Quantum Approximate Optimization Algorithm (QAOA) [1], which is one of the great quantum gate model algorithms for designing novel approximation-based optimization algorithms.

III. Quantum Approximate Optimization Algorithm (QAOA)

QAOA which was introduced recently is a quantum gate model algorithm to solve combinatorial optimization problems [1]. Unlike quantum annealing machines those are solvers [13], QAOA does not need empirical characterization. In addition, the performance of quantum annealing machines can be decreased while execution time increases [14]. On the other hand, the performance of the p-level QAOA (QAOA$_p$) increases continually along with p. Theoretically, the approximation ratio r of QAOA$_p$ reaches 1 at ∞-level, i.e., $r \to 1$ when $p \to \infty$ [15]. Furthermore, one additional benefits of QAOA is fundamentally based on its simple structure. This simplicity leads the use of QAOA on Noisy Intermediate-Scale Quantum (NISQ) devices [16]. In this section, We explain the overview of QAOA which has these advantages.

A. The overview of QAOA

This section presents definitions those are used in QAOA explanations. Note that the following definitions are based on the original ideas by *Farhi et al.* [1] and also discussed in [5], [15], [17], [18].

Definition 1 (Objective Function)**.** *In a combinatorial optimization problem defined on n-bit binary strings z, the objective function is defined as follows:*

$$f(z) : \{0,1\}^n \to R. \tag{2}$$

Definition 2 (Phase Operators)**.** *We can map the objective function* (2) *to the phase Hamiltonian, thus finding the optimal value of the objective function is a special case of finding the extremal eigenvalues for the phase Hamiltonian. The phase Hamiltonian H_P encodes the objective function f and acts diagonally on the computational basis states of 2^n dimensional Hilbert space (n-qubit space).*

$$H_P |z\rangle = f(z) |z\rangle . \tag{3}$$

In addition, the phase operators are defined as follows:

$$U_P(\gamma) = e^{-i\gamma H_P}, \tag{4}$$

where γ is a parameter.

Definition 3 (Mixing Operators)**.** *The mixing Hamiltonian H_M is defined as follows:*

$$H_M = \sum_{j=1}^{n} \sigma_j^x, \tag{5}$$

where σ_j^x is the Pauli-X operator and n is identical to the n in (2)*. In the quantum mechanical systems, the Pauli-X operator acts as the NOT operator, i.e., $\sigma_j^x |1\rangle = |0\rangle$ and $\sigma_j^x |0\rangle = |1\rangle$. In addition, the mixing operators are defined as follows:*

$$U_M(\beta) = e^{-i\beta H_M}, \tag{6}$$

where β is a parameter.

Definition 4 (Initial State)**.** *The initial state is as follows according to the superposition principle:*

$$|s\rangle = |+\rangle^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_z |z\rangle . \tag{7}$$

Based on the definitions above, we can define the state of the p-level QAOA by applying the phase operator and the mixing

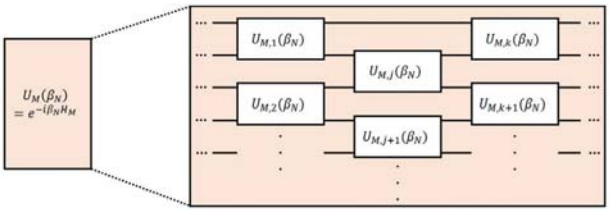

Fig. 2: Quantum Alternating Operator Ansatz circuit [5]. This figure illustrates the details of $U_M(\beta_N)$ in Fig. 1.

operator alternately, as follows:

$$|\gamma, \beta\rangle = U_M(\beta_p)U_P(\gamma_p)\cdots U_M(\beta_1)U_P(\gamma_1)\,|s\rangle\,, \qquad (8)$$

with an integer $p \geq 1$ and $2p$ parameters $\gamma_1 \cdots \gamma_p \equiv \gamma$ and $\beta_1 \cdots \beta_p \equiv \beta$ [1]. After the measurements in the computational basis are repeatedly performed in this state (8), the expectation value of H_P can be obtained as follows:

$$\langle H_P\rangle := \langle\gamma, \beta|H_P|\gamma, \beta\rangle = \langle f\rangle_{(\gamma,\beta)}\,, \qquad (9)$$

where $\langle f\rangle$ is the expectation value of the objective function (2) [5]. As shown in Fig. 1, the maximum or minimum value of $\langle H_P\rangle$ can be obtained by repeating the process of finding the optimal values of the parameters γ and β. The iterative process for finding optimal parameters uses classical optimization methods. For this reason, QAOA is in the category of hybrid quantum-classical algorithms [16], [18]–[20].

In following subsection, we discuss about the approximation ratio of QAOA$_p$.

B. The approximation ratio of QAOA$_p$

Suppose that there is a combinatorial optimization problem that needs to maximize the objective function f. In this problem, we can obtain the optimal parameters γ_{op} and β_{op} via QAOA$_p$, as illustrated in in Fig. 1. Based on (1), the approximation ratio of QAOA$_p$ for this problem is as follows:

$$r = \frac{\langle f\rangle_{(\gamma_{op},\beta_{op})}}{f_{max}} = \frac{\langle\gamma_{op}, \beta_{op}|H_P|\gamma_{op}, \beta_{op}\rangle}{f_{max}}. \qquad (10)$$

Farhi et al. [1] proved that the 1-level QAOA (QAOA$_1$) achieves $R \geq 0.6924$ $(= r)$ for MaxCut on unweighted 3-regular graphs. According to this result, QAOA$_p$ has good performance at $p = 1$. However, there are little research results for the QAOA$_p$ at $p \gg 1$. In this regard, designing the appropriate parameter setting strategy is one of important research topics [21]. In the following subsection, we briefly discuss the concept and advantage of the Quantum Alternating Operator Ansatz [5], which is the extension of QAOA [1].

C. Quantum Alternating Operator Ansatz

The Quantum Alternating Operator Ansatz is an extension of QAOA which is useful for various problems with constraints [5], [17]. The key to this extension is about to enable the alternation using partial mixing operators and phase-separation operators. As shown in Fig. 2, the mixing operator family $U_M(\beta)$ of QAOA can be decomposed into a sequence of partial mixing operators $U_{M,\zeta}(\beta)$. These partial mixing operators make the QAOA mapping of the general problems be more convenient. This extension to the Quantum Alternating Operator Ansatz is very significant because it increases the flexibility of QAOA for various applications and it is also beneficial in terms of implementation.

TABLE I: Mapping the relationships between Boolean clauses and Hamiltonians [17] where the I and Z stand for identity matrix and Pauli-Z matrix, respectively.

$f(x)$	H_f
x	$\frac{1}{2}I - \frac{1}{2}Z$
$\bar{x}$	$\frac{1}{2}I + \frac{1}{2}Z$
x_1x_2	$\frac{1}{4}(I - Z_1 - Z_2 + Z_1Z_2)$
$\bigwedge_{j=1}^{k} x_j$	$\frac{1}{2^k}\prod_j(1 - Z_j)$
$x_1 \vee x_2$	$\frac{3}{4}I - \frac{1}{4}(Z_1 + Z_2 + Z_1Z_2)$
$\bigvee_{j=1}^{k} x_j$	$1 - \frac{1}{2^k}\prod_j(1 + Z_j)$
$x_1 \oplus x_2$	$\frac{1}{2}I - \frac{1}{2}Z_1Z_2$
$x_1 \Rightarrow x_2$	$\frac{3}{4}I + \frac{1}{4}(Z_1 - Z_2 + Z_1Z_2)$

IV. Applications

This section presents the QAOA mapping for maximum cut (MaxCut) problem and the max-independent set (MIS) problem. For the mapping methodologies in the two problems, we referred to methods such as *Farhi et al.* [1] and *Hadfield et al.* [5], [17].

A. Maximum Cut

Problem 1 (Maximum Cut (MaxCut))**.** *Given a graph $G = (V, E)$ with $|V| = n$ vertices and $|E| = m$ edges, maximize the number of edges crossing the cut. The cut separates V_1 and V_2, i.e., one vertex of the edges crossing the cut should be in V_1 and the other vertex should be in V_2, where $V_1 \subset V$, $V_1 \cup V_2 = V$, and $V_1 \cap V_2 = \emptyset$.*

MaxCut is one of the fundamental cases in QAOA applications. In an n-vertex graph, states can be represented by n qubits, i.e., one vertex per one qubit can be expressed. Thus, n-vertex graphs can have 2^n partitions of V and 2^n computational basis states encode all possible partitions. It is unconstrained binary combinatorial optimization because all states are feasible.

Component 1. 1 (Configuration Space of MaxCut)**.** *The set of n-bit strings $x = x_1x_2\cdots x_n$, where $x_i = 1$ if vertex $i \in V_1$.*

Component 1. 2 (Objective Function of MaxCut)**.** *The two vertices of the edge must be opposite to each other. Therefore, the objective function is as follows:*

$$f(x) = \sum_{(uv)\in E} (x_u \oplus x_v), \qquad (11)$$

where $u \in V_1$ and $v \in V_2$.

Component 1. 3 (Phase Hamiltonian of MaxCut)**.** *The objective function (11) corresponds to $f(x) = x_1 \oplus x_2$ in Table I, thus the phase Hamiltonian is as follows:*

$$H_P = \sum_{(uv)\in E} \frac{1}{2}(I - Z_u Z_v). \tag{12}$$

Other components about QAOA mapping on MaxCut are omitted because they don't need a specific technical approach. With the phase Hamiltonian (12) obtained above, we can obtain remaining components smoothly as defined in Sec. III-A.

B. Max-Independent Set

Problem 2 (Max-Independent Set (MIS))**.** *Given a graph $G = (V, E)$, with $|V| = n$ vertices and $|E| = m$ edges, maximize independent subset $V^* \subseteq V$ such that no two vertices in V^* are adjacent.*

MIS is a kind of constrained binary optimization problems, unlike MaxCut. There are no polynomial-time classical algorithms for MIS unless $P = NP$ [22]. Thus, we discuss QAOA mapping briefly on MIS in this subsection.

Component 2. 1 (Configuration Space of MIS)**.** *The set of n-bit strings $x = x_1 x_2 \cdots x_n$ representing V^* where $x_i = 1 \Leftrightarrow i \in V^*$.*

Component 2. 2 (Objective Function of MIS)**.** *The objective function is defined as follows:*

$$f(x) = \sum_{j=1}^{n} x_j, \tag{13}$$

where $f(x)$ is the number of vertices in V^*.

Component 2. 3 (Phase Hamiltonian of MIS)**.** *The objective function (13) corresponds to $f(x) = x$ in Table I, thus the phase Hamiltonian is as follows:*

$$H_P = \sum_{u\in V} \frac{1}{2}(I - Z_u) = \frac{n}{2} I - \frac{1}{2} \sum_{u\in V} Z_u. \tag{14}$$

Component 2. 4 (Phase Operator of MIS)**.** *If we remove a constant in (14), the phase operator can be as follows:*

$$U_P(\gamma) = e^{i\frac{\gamma}{2}\sum_{u\in V} Z_u} = \sum_{u\in V} e^{i\frac{\gamma}{2} Z_u}. \tag{15}$$

Component 2. 5 (Mixing Hamiltonian of MIS)**.** *If there is a vertex $w \notin V^*$ and all adjacent vertices of w do not belong to V^*, w should be included in V^*. In other words, if the adjacent vertices $v_i, \cdots, v_\alpha \notin V^*$, then $x_w = 1$, i.e., flip the bit x_w if $\bar{x_{v_1}} \cdots \bar{x_{v_\alpha}} = 1$. According to this rule and the case of $f(x) = \bar{x}$ in Table I, the mixing Hamiltonian for w can be expressed as follows:*

$$H_{M,w} = (\bar{x_{v_1}} \cdots \bar{x_{v_\alpha}}) \cdot X_w = \frac{1}{2^\alpha} X_w \prod_{j=1}^{\alpha} (I + Z_{v_j}), \tag{16}$$

where X stands for Pauli-X matrix for bit-flip. Note that the generalized form of (16) for vertex $u \in V$ can be presented as follows:

$$H_{M,u} = \frac{1}{2^\alpha} X_u \prod_{j=1}^{\alpha} (I + Z_{v_j}). \tag{17}$$

Component 2. 6 (Mixing Operator of MIS)**.** *The basic mixing operator from (17) is as follows:*

$$U_{M,u}(\beta) = e^{-i\beta H_{M,u}}. \tag{18}$$

According to Hadfield-style extension, the following two mixers can be defined:

$$U_M^{(H)}(\beta) = e^{-i\beta \sum_u H_{M,u}}, \tag{19}$$

$$U_M(\beta) = \prod_{u\in V} U_{M,u}(\beta), \tag{20}$$

where (19) *and* (20) *are the simultaneous controlled-X mixer and the sequential controlled-X mixer, respectively.*

Component 2. 7 (Initial State of MIS)**.** *For the initial state, we recommend the following state:*

$$|s\rangle = |0\rangle^{\otimes n}, \tag{21}$$

which is the simplest state, i.e., $V^* = \emptyset$.

In this subsection, we studied the components of QAOA mapping on MIS. We discussed QAOA mapping on MIS with the algorithmic viewpoint instead of the implementation viewpoint. Based on this result, it can be realized that constrained optimization problems can be mapped using the concept of the Quantum Alternating Operator Ansatz.

V. Concluding Remarks and Future Work

In summary, we have studied QAOA and its applications to two combinatorial optimization problems, i.e., MaxCut and MIS. We have also studied from the approximation algorithm to the Quantum Alternating Operator Ansatz for the fundamental understanding of QAOA. We have confirmed the structural superiority and the scalability of QAOA with mathematical and physical approaches to the core ideas of QAOA such as phase Hamiltonian, mixing Hamiltonian, and so forth. The QAOA mapping examples for both MaxCut and MIS have also demonstrated the applicability of QAOA to various areas such as networks and machine learning.

As future work directions, perfect solutions about parameter-determination issues and implementation issues are expected. These research results will lead us to pioneer new areas of QAOA applications.

Acknowledgment

This research was supported by National Research Foundation of Korea (2017R1A4A1015675, 2019R1A2C4070663). J. Kim is the corresponding author of this paper (e-mail: joongheon@korea.ac.kr).

REFERENCES

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

[2] Z.-C. Yang, A. Rahmani, A. Shabani, H. Neven, and C. Chamon, "Optimizing variational quantum algorithms using pontryagin's minimum principle," *Physical Review X*, vol. 7, no. 2, p. 021027, 2017.

[3] D. Venturelli, M. Do, E. Rieffel, and J. Frank, "Compiling quantum circuits to realistic hardware architectures using temporal planners," *Quantum Science and Technology*, vol. 3, no. 2, p. 025004, 2018.

[4] A. Oddi and R. Rasconi, "Greedy randomized search for scalable compilation of quantum circuits," in *International Conference on the Integration of Constraint Programming, Artificial Intelligence, and Operations Research*. Springer, 2018, pp. 446–461.

[5] S. Hadfield, Z. Wang, B. O'Gorman, E. G. Rieffel, D. Venturelli, and R. Biswas, "From the quantum approximate optimization algorithm to a quantum alternating operator ansatz," *Algorithms*, vol. 12, no. 2, p. 34, 2019.

[6] E. Anschuetz, J. Olson, A. Aspuru-Guzik, and Y. Cao, "Variational quantum factoring," in *International Workshop on Quantum Technology and Optimization Problems*. Springer, 2019, pp. 74–85.

[7] G. Verdon, J. Pye, and M. Broughton, "A universal training algorithm for quantum deep learning," *arXiv preprint arXiv:1806.09729*, 2018.

[8] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, and M. Protasi, *Complexity and approximation: Combinatorial optimization problems and their approximability properties*. Springer Science & Business Media, 2012.

[9] V. V. Vazirani, *Approximation algorithms*. Springer Science & Business Media, 2013.

[10] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy, "Proof verification and the hardness of approximation problems," *Journal of the ACM (JACM)*, vol. 45, no. 3, pp. 501–555, 1998.

[11] J. Håstad, "Some optimal inapproximability results," *Journal of the ACM (JACM)*, vol. 48, no. 4, pp. 798–859, 2001.

[12] M. X. Goemans and D. P. Williamson, "Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming," *Journal of the ACM (JACM)*, vol. 42, no. 6, pp. 1115–1145, 1995.

[13] C. C. McGeoch, "Adiabatic quantum computation and quantum annealing: Theory and practice," *Synthesis Lectures on Quantum Computing*, vol. 5, no. 2, pp. 1–93, 2014.

[14] E. Crosson, E. Farhi, C. Y.-Y. Lin, H.-H. Lin, and P. Shor, "Different strategies for optimization using the quantum adiabatic algorithm," *arXiv preprint arXiv:1401.7320*, 2014.

[15] 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," *arXiv preprint arXiv:1812.01041*, 2018.

[16] J. Preskill, "Quantum computing in the nisq era and beyond," *Quantum*, vol. 2, p. 79, 2018.

[17] S. Hadfield, "Quantum algorithms for scientific computing and approximate optimization," *arXiv preprint arXiv:1805.03265*, 2018.

[18] G. G. Guerreschi and A. Matsuura, "Qaoa for max-cut requires hundreds of qubits for quantum speed-up," *Scientific reports*, vol. 9, no. 1, p. 6903, 2019.

[19] R. Shaydulin, H. Ushijima-Mwesigwa, C. F. Negre, I. Safro, S. M. Mniszewski, and Y. Alexeev, "A hybrid approach for solving optimization problems on small quantum computers," *Computer*, vol. 52, no. 6, pp. 18–26, 2019.

[20] E. G. Rieffel, S. Hadfield, T. Hogg, S. Mandrà, J. Marshall, G. Mossi, B. O'Gorman, E. Plamadeala, N. M. Tubman, D. Venturelli *et al.*, "From ans\" atze to z-gates: a nasa view of quantum computing," *arXiv preprint arXiv:1905.02860*, 2019.

[21] Z. Wang, S. Hadfield, Z. Jiang, and E. G. Rieffel, "Quantum approximate optimization algorithm for maxcut: A fermionic view," *Physical Review A*, vol. 97, no. 2, p. 022304, 2018.

[22] L. Trevisan, "Inapproximability of combinatorial optimization problems," *arXiv preprint cs/0409043*, 2004.