Quantum Alternating Operator Ansatz in TSP problems

August 2019

1 The Quantum Alternating Operator Ansatz

An instance of a optimization problem is a pair (F, f) where F is the domain, also called the set of feasible points, and $f : F \to \mathbb{R}$ is the objective function to be optimized. Let $\mathcal{F}$ be the Hilbert space of dimension $|F|$, whose standard basis we take to be $\{|\mathbf{x}\rangle : \mathbf{x} \in F\}$. A generalized quantum approximate optimization algorithm(QAOA)[1] circuit is characterized by two parameterized families of operators on $\mathcal{F}$:

- A family of phase-separation operators $U_P(\gamma)$ that depends on the objective function f, and;
- A family of mixing operators $U_M(\beta)$ that depends on the domain and its structure,

where β and γ are real parameters. Specifically, a QAOA circuit consist of p alternating applications of operators from these two families:

$$Q_p(\boldsymbol{\beta}, \boldsymbol{\gamma}) \equiv U_M(\beta_p)U_P(\gamma_p) \cdots U_M(\beta_1)U_P(\gamma_1). \tag{1}$$

This quantum alternating operator ansatz(QAOA)[2] consists of the states representable as the application of such a circuit to a suitably simple initial state $|s\rangle$:

$$|\boldsymbol{\beta}, \boldsymbol{\gamma}\rangle \equiv Q_p(\boldsymbol{\beta}, \boldsymbol{\gamma}) |s\rangle . \tag{2}$$

For any function f, not just an objective (cost) function, we define H_f to be the quantum Hamiltonian that acts as f on basis states as:

$$H_f |\mathbf{x}\rangle = f(\mathbf{x}) |\mathbf{x}\rangle . \tag{3}$$

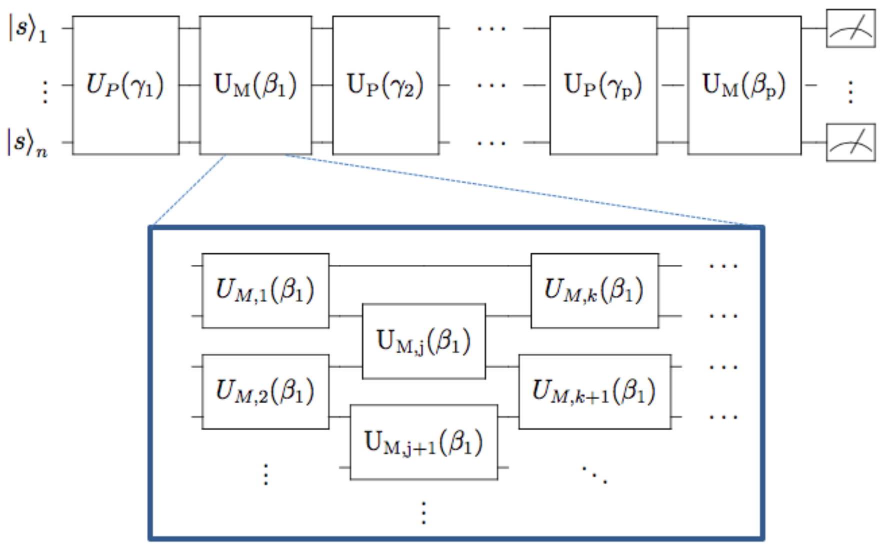

Figure 1: The quantum alternating operator ansatz quantum circuit schematic. The box shows an example decomposition of a QAOA mixing operator family $U_M(\beta)$ into a sequence of partial mixers $U_{M,\alpha}(\beta)$.

An overall quantum circuit schematic for a QAOA mapping is presented in Figure 1. For a given optimization problem, a QAOA mapping of a problem consists of a family of phase-separation operators, a family of mixing operators, and a starting state. The design criteria for the three components of QAOA mapping of a problem is listed in the following:

- **Initial state.** The QAOA mapping scheme requires the initial state $|s\rangle$ is trivial to implement, which means that it can be implemented by a constant-depth(in the size of the problem) quantum circuit from the $|0...0\rangle$ state. Often, the scheme takes as the initial state a single feasible solution, usually implementable by a depth-1 circuit consisting of single-qubit bit-flip operations X.(this criterion could be relaxed to logarithmic depth if needed.)

- **Phase-separation unitaries.** The scheme requires the family of phase-separation operators to be diagonal in the computational basis.

- **Mixing unitaries.** The scheme requires the family of mixing operators $U_M(\beta)$ to:

- Preserve the feasible subspace: For all values of the parameter β, the resulting unitary takes feasible states to feasible states. For instance, the computational-basis states of the feasible subspace have the same Hamming weight in some combinator optimization question using the one-hot encoding. If this is the case, the mixers should preserve Hamming weight.
- Provide transitions between all pairs of states corresponding to feasible points. More concretely, for any pair of feasible computational-basis states $\mathbf{x}, \mathbf{y} \in F$, there is some parameter value β^* and some integer r such that the corresponding mixer connects those two states: $|\langle \mathbf{x}| U_M^r(\beta^*) |\mathbf{y}\rangle| > 0$.

2 The TSP Problem Using the QAOA Scheme

Problem. Given a set of n cities, and distances $d : [n]^2 \to \mathbb{R}_+$, find an ordering of the cities that minimizes the total distance traveled for the corresponding tour. A tour visits each city exactly once and returns from the last city to the first. Note that we defined $[n] = \{1, 2, \ldots, n\}$ and $[0, n] = \{0, 1, \ldots, n\}$.

2.1 Mapping

The configuration space here is the set of all orderings of the cities. Labeling the cities by $[n]$, the ordering $\boldsymbol{\iota} = (\iota_1, \iota_2, \ldots, \iota_{n-1}, \iota_n)$ indicates traveling from city ι_1 to city ι_2, then on to city ι_3 and so on until finally returning from city ι_n back to city ι_1. The configuration space includes some degeneracy in solutions with respect to cyclic permutations; specifically, for any ordering $\boldsymbol{\iota}$, the configuration space includes both $(\iota_1, \iota_2, \ldots, \iota_{n-1}, \iota_n)$ and $(\iota_2, \iota_3, \ldots, \iota_n, \iota_1)$, even though they are essentially the same solution to the TSP. We leave in this degeneracy in the constructions of this section in order to preserve symmetries which make it simpler to construct and present our mixers. Note that, in practice, this degeneracy may be removed by fixing a particular city as the starting point.

As there are no problem constraints, the domain is the same as the configuration space. The objective function is:

$$f(\boldsymbol{\iota}) = \sum_{j=1}^{n} d_{\iota_j,\iota_{j+1}}, \tag{4}$$

where we employed the convention $\iota_{n+1} := \iota_1$.

Ordering swap partial mixing Hamiltonians. Consider $\{\iota_i, \iota_j\} = \{u, v\}$, indicating that city u (resp. v) is visited at the ith (resp. jth) stop on the tour, or vice versa. The value-selective ordering swap mixing Hamiltonians, $H_{\mathrm{PS},\{i,j\},\{u,v\}}$, which swap the ith and jth elements in the ordering if and only if those elements are the cities u and v:

$$\begin{aligned} &H_{\mathrm{PS},\{i,j\},\{u,v\}} \\ &= \sum_{\boldsymbol{\iota}:\{\iota_i,\iota_j\}=\{u,v\}} \left|(\iota_1,\ldots,\iota_{i-1},v,\ldots\iota_{j-1},u,\ldots\iota_n)\right\rangle\left\langle(\iota_1,\ldots,\iota_{i-1},u,\ldots\iota_{j-1},v,\ldots\iota_n)\right| . \end{aligned} \tag{5}$$

Also, denoting the adjacent ordering swap mixing Hamiltonians as

$$H_{\mathrm{PS},i,\{u,v\}} = H_{\mathrm{PS},\{i,i+1\},\{u,v\}} . \tag{6}$$

To swap the ith and jth elements of the ordering regardless of which cities those are, we used the value-independent ordering swap partial mixing Hamiltonian:

$$H_{\mathrm{PS},\{i,j\}} = \sum_{\{u,v\}\in\binom{[n]}{2}} H_{\mathrm{PS},\{i,j\},\{u,v\}}. \tag{7}$$

Denoting the adjacent value-independent ordering swap partial mixing Hamiltonians as

$$H_{\mathrm{PS},i} = \sum_{\boldsymbol{\iota}} \left|(\iota_1,\ldots\iota_{i-1},\iota_{i+1},\iota_i,\iota_{i+2},\ldots,\iota_n)\right\rangle\left\langle(\iota_1,\ldots,\iota_n)\right|, \tag{8}$$

which swap the ith element with the subsequent one regardless of which cities those are.

These partial mixers can be combined in several ways to form full mixers, of which we explore two types.

Simultaneous ordering swap mixer. Defining $H_{\mathrm{PS}} = \sum_{i=1}^{n} H_{\mathrm{PS},i}$, we have the "simultaneous ordering swap mixer":

$$U_{\text{sim-PS}}(\beta) = e^{-i\beta H_{\mathrm{PS}}}. \tag{9}$$

Color-parity ordering swap mixer. To define the ordered partition, we first defined an ordered partition on the set of adjacent partial mixers $U_{\mathrm{PS},i,\{u,v\}}$ for a fixed tour position i, where the parts of this partition contains mutually commuting partial mixers. We then partitioned the i to obtain a full ordered partition. Two partial mixers $U_{\mathrm{PS},i,\{u,v\}}$ and $U_{\mathrm{PS},i,\{u',v'\}}$ commute as long as $\{u,v\} \cap \{u',v'\} = \emptyset$. Partitioning the $\binom{n}{2}$ pairs of cities into κ parts such that each part contains only mutually disjoint pairs is equivalent to considering a κ-edge-coloring of the complete graph K_n and assigning an ordering to the colors. For odd n, $\kappa = n$ suffices, and for even n, $\kappa = n-1$ suffices. Let $\mathcal{P}_{\mathrm{col}} = (P_1, \ldots, P_c, \ldots, P_\kappa)$ be the resulting ordered partition, which we call a "color partition" of the pairs of cities. For example, for $U_n = 4$, the partition is $\mathcal{P}_{\mathrm{col}} = (\{\{1,2\},\{3,4\}\},\{\{1,3\},\{2,4\}\},\{\{1,4\},\{2,3\}\})$. For different tour positions i, two partial unitaries $U_{\mathrm{PS},i,\{u,v\}}$ and $U_{\mathrm{PS},i',\{u',v'\}}$ commute if i and i' are not consecutive ($|i-i' \mod n| > 1$). Thus, for partitioning the positions, we may use the parity partition $\mathcal{P}_{\mathrm{par}}$. We can thus define the "color-parity" ordered partition $\mathcal{P}_{\mathrm{CP}} = \mathcal{P}_{\mathrm{col}} \times \mathcal{P}_{\mathrm{par}}$, with the induced lexicographical ordering of the parts. The part $P_{c,\mathrm{odd}}$ contains all $U_{\mathrm{PS},i,\{u,v\}}$ such that i is odd and edge $\{u,v\}$ is colored c, i.e., in P_c, and defines the unitary:

$$U_{c,\mathrm{odd}}(\beta) = \prod_{(i,\{u,v\}) \in P_{c,\mathrm{odd}}} U_{\mathrm{PS},i,\{u,v\}}(\beta), \tag{10}$$

where the ordering of the products does not matter because each term commutes. It is a similar case for $P_{c,\mathrm{even}}$ and $U_{c,\mathrm{last}}$, and $P_{c,\mathrm{last}}$ and $U_{c,U}$. Thus, we have the full color-parity mixer:

$$U_{\mathrm{CP}}(\beta) = U_{\mathcal{P}_{\mathrm{CP}}\text{-PC}}(\beta) = \prod_{\mathrm{P}_{c,\pi} \in \mathcal{P}_{\mathrm{CP}}} U_{c,\pi}, \tag{11}$$

where the unitaries $\{U_{c,\pi}\}$ are applied in the order they appear in $\mathcal{P}_{\mathrm{CP}}$. The color-parity partition is optimal with respect to the number of parts in the partition (exactly so for even n and up to an additive factor of 2 for odd n). By construction, application of this mixer to any feasible state results in a feasible state, thus satisfying the first design criterion. With regard to the second criterion, while a single application of this mixer will have nonzero transitions only between orderings that swap cities in tour positions no more than two apart, repeating the mixer sufficiently many times results in nonzero transitions between any two states representing orderings.

2.1.1 Compilation

Encoding orderings. we encoded orderings in two stages: First into strings, and then into bits. In direct encoding, an ordering $\boldsymbol{\iota} = (\iota_1, \ldots, \iota_n)$ is encoded directly as a string $[n]^n$ of integers. Then we applied the one-hot encoding with n^2 binary variables; the binary variable $x_{j,u}$ indicates whether or not $\iota_j = u$ in the ordering, in other words, whether city u is visited at the j-th stop of the tour.

Phase separator. We used the phase function $g(\boldsymbol{\iota}) = 4f(\boldsymbol{\iota}) - (n-2)\sum_{u=1}^{n}\sum_{v=1}^{n} d(u,v)$, which translates to a phase separator encoded as:

$$H_P^{\text{enc}} = \sum_{i=1}^{n}\sum_{u=1}^{n}\sum_{v=1}^{n} d(u,v) Z_{u,i} Z_{v,i+1}. \tag{12}$$

Mixer. The individual value-selective ordering swap partial mixer, which swaps cities u, v between tour positions i and j, is expressed in the one-hot encoding as:

$$U^{\text{enc}}_{\text{PS},\{i,j\},\{u,v\}}(\beta) = e^{\beta H_{\text{PS},\{i,j\},\{u,v\}}}\ , \tag{13}$$

$$H^{\text{enc}}_{\text{PS},\{i,j\},\{u,v\}} = S^+_{u,i} S^+_{v,j} S^-_{u,j} S^-_{v,i} + S^- S^-_{u,i} S^-_{v,j} S^+_{u,j} S^+_{v,i}, \tag{14}$$

where:

$$S^+ = X + iY = |1\rangle\langle 0|\,, \tag{15}$$

$$S^- = X - iY = |0\rangle\langle 1|\,. \tag{16}$$

The ith adjacent value-selective swap partial mixer (Equation (6)) is the special case:

$$H^{\text{enc}}_{\text{PS},i,\{u,v\}} = S^+_{u,i} S_{v,i+1} S^-_{u,i+1} S^+_{v,i} + S^-_{u,i} S^-_{v,i+1} S^+_{u,i+1} S^+_{v,i}\ . \tag{17}$$

Each of the two terms of the form $S^+S^+S^-S^-$ in Equation (17), can be written as a sum of eight terms, each a product of 4 Pauli operators (e.g., $XXYY$).

Initial state. The initial state, an arbitrary ordering, can be prepared from the zero state $|00\ldots 0\rangle$ using at most n single-qubit X gates.

References

[1] F. Edward, G. Jeffrey, and G. Sam. A quantum approximate optimization algorithm. arXiv:1411.4028, 2014.

[2] H. Stuart, Zhihui W, Bryan O, R. Eleanor, V. Davide, and B. Rupak. From the quantum approximate optimization algorithm to a quantum alternating operator ansatz. Algorithms, 12(2):34, 2019.