Title: Novel Solutions to Airline Scheduling Problems: From Classical to Quantum

1 Introduction

Airlines face several large-scale planning and scheduling problems on a regular basis. At the basic level, the problem is, given customer demand on a daily and weekly basis for different flight segments, to assign aircrafts and crew to these segments, and to decide the order in which flight segments will be served. The objective is typically to minimize a combination of time and resources required to service the total demand. These would translate into good quality of service and profit maximization respectively. In addition to the technical challenge posed by the massive scale of these problems, additional challenges are posed by practical considerations such as matching between crew, aircrafts and flight segments, ordering of flights (e.g., a popular domestic flight scheduled after an incoming international flight), refueling/maintenance times for aircrafts and resting time for crews. It is of interest to design algorithms to solve these complex problems with reasonable accuracy and with small computational times. High inaccuracy could lead to inefficient use of resources, whereas large computational times could lead to huge overhead costs for the airline in the form of computational resources. Indeed, fast computation might become a hard constraint when responding to disruptions, e.g., as brought about by the pandemic.

Many airline problems in a real-world have numerous, complex rules so that it makes a problem hard to solve. Sometimes, it is even hard to find a feasible solution and also hard to be formulated as a single model. For instance, crew scheduling problem is generally divided into two sub problems, crew pairing problem (CPP) and crew rostering problem (CRP) and still a heuristic method is used for solving a daily crew rostering problem [1]. However, ideally, it should be one integrated problem and model.

Quantum computers have the potential to be game changers in computing paradigm, with the possibility of solving large-scale problems which would otherwise be impossible to solve even on supercomputers. While quantum computing algorithms have attracted interest at least for three decades, it is only recently that the hardware has reached maturity to facilitate implementation. It is anticipated that, in the near term, generic heuristic algorithms such as the quantum approximate optimization algorithm (QAOA) [2] are most likely to be implemented. QAOA is a hybrid classical-quantum algorithm for a class of combinatorial optimization problems. Consequently, there has been some preliminary work on applying this algorithm to basic instances of airline scheduling problems, e.g., see [3].

The quantum hardware that is expected to be conveniently accessible in the near term will be limited. It is therefore of interest to understand the tradeoffs between scale, accuracy, and computation time for complex airline problems on such limited hardware. We are also interested in a generalization of QAOA similar in spirit to [4]. In this approach, the challenging component of optimization is solved *offline* by constructing a map from problem input parameters, e.g., availability of crew/aircraft and customer demand, to subset of optimal decision variables in the form of a neural network, through classical optimization techniques. The remaining decision variables are solved *online* using QAOA. The objective is to optimize the usage of limited quantum hardware to solve problems in real-time, e.g., as would be necessitated during disruptions. For the longer term, when more resources and a variety of quantum hardware are expected to become accessible, it is of interest to investigate specialized algorithms which explore the structure of airline scheduling problems. Towards that purpose, we investigate quantum graph optimization approach, which is known to beat classical approaches in some instances of minimum spanning tree problem [5].

2 Objectives

The vision of this project is to lay foundations for algorithms, analysis and simulation code for practical airline scheduling problems using a combination of classical and computing paradigms that are best suited for hardware availability in the near term as well as the long term. The focus is on baseline assignment for nominal operating conditions as well as to enable quick response to disruptions. Towards these goals, the key objectives are:

1. Formulate practical airline scheduling problems as variants of the multi-agent Eulerian path (MAEP) problem. Identify two formulations, MAEP-1 and MAEP-2, for priority.
2. Approximation algorithms for MAEP problems using classical paradigm.
3. Quantum algorithms for MAEP problems using QAOA paradigm and Quantum Graph Optimization Approach (QGOA).
4. Implementation of classical algorithms on classical computer.
5. Implementation of quantum algorithms *by simulation on classical computer* using Qiskit [6], and *on a quantum device* using IBM Open Quantum [7].
6. Partitioning variables for online-offline computation for quick response to disruptions.

Note that because quantum computing must ultimately be executed on a physical quantum device, there is some inherent error in conversion between logical and physical implementation. However, correction of such quantum errors will not be the focus of this work. We will assume standard quantum error correcting practices, and use repetition codes available from IBM and QisKit to mitigate these effects [8].

3 Technical Approach

A rough outline of our technical approach is as follows. We construct a directed *demand* graph whose nodes are airports and edges are flight segments. Edge weight is (daily) flight frequency. A multi-agent Eulerian path (MAEP) problem is to schedule "agents" so that each edge gets traversed as many times as its weight. An agent is a combination of aircraft and crew. We shall solve the minimum time MAEP under practical constraints such as matching aircraft type/crew size with segments, downtime of aircraft/crew (rest, refueling, maintenance), sequencing of segments (e.g., LAX to SFO after ICN to LAX).

The matching constraint is reminiscent of routing heterogeneous vehicles to service heterogeneous demand, e.g., see [9], where approximation algorithms were designed based on "team-forming" and optimal service order for each team. The airline context presents more challenges in the form of *temporal* constraints. For example, multiple flights in the same segment should not be clustered together in time, refueling/maintenance time of aircrafts, rest time for crew, etc. We shall formulate and provide classical solutions to two representative variants of MAEP problems with these constraints.

The Quantum Approximate Optimization Algorithm (QAOA) [2] is a generic approach for several combinatorial optimization problems, including the tail assignment problem [3]. On the other hand, one could developed specialized quantum algorithms by leveraging tools developed for graph problems, e.g., see [5]. We plan to investigate the potential of these approaches for the two chosen variants of MAEP, and benchmark against classical algorithms. Our methodology will enable pre-computing solutions to a large number of sample demand scenarios at reasonable computational cost. This allows better training of neural network models to interpolate between the samples, and hence provide accurate response to generic disruptions, similar in spirit to [4].

The methodological development will be supplemented with simulation implementation on classical and quantum computers, subject to availability of corresponding resources.

Airline Scheduling & Routing as Eulerian Path Problem

Consider a simple scenario illustrated in Figure 1. LAX has two inbound flights from, one from ICN and the other from SFO, and two outbound flights JFK. Assume an incoming international flight needs at least one hour downtime for rest, refueling and maintenance, then the constraints should ensure that the flight ICN-LAX (0800) is not connected with LAX-JFK (0855). In addition, there are matching constraints between aircraft/crew and flight segments. The complexity increases dramatically with the size of operations.

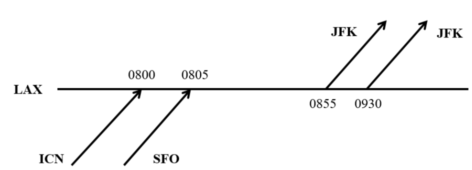

Figure 1: Aircraft routing problem

Consider the following simple abstraction of demand and supply for an airline network $G = (V, E)$, which is a directed multi-graph, i.e., it allows multiple link between the same pair of nodes. This multiplicity represents the demand frequency on the corresponding segment. The node set V consists of the destinations. A directed link exists is present in the link set E if a flight segment needs to be served in that direction. Let $c(G')$ denote the minimum cost Eulerian path on the sub-graph G' (if it exists). Let m be the number of aircrafts available to serve the demand. We are interested in solving the following daily assignment problem:

$$\min_{G^{(1)},\ldots,G^{(m)}} \max_{j\in\{1,\ldots,m\}} c(G^{(i)}) \quad \text{s.t.} \quad G^{(1)} \cup \ldots \cup G^{(m)} = G \qquad \text{(MAEP-0)}$$

(MAEP-0) captures only the basic features of airline scheduling problems. We plan to consider the following two generalizations as representative of practical considerations:

- MAEP-1: augment (MAEP-0) with crews and include constraints on matching aircrafts/crew to flight segments. This naturally addresses heterogeneity in attributes of different aircrafts and skillsets of crews.
- MAEP-2: augment (MAEP-0) with temporal constraints such as refueling time for aircrafts, resting time for crew, and preference in ordering of flights. The latter, e.g., can enforce schedule of a popular domestic segment (e.g., LAX to SFO) after the incoming international flight (e.g., ICN to LAX).

For the purpose of illustration, and also since classical algorithms do not exist even for the basic version, we describe our methodologies for the basic version (MAEP-0).

3.1 Classical Solution Approach

The *single-agent* version of the Eulerian path problem finds application in a variety of domains from scheduling to DNA sequencing [10]. Hence, it has attracted great research interest in terms of classical algorithms. However, surprisingly, its multi-agent counterpart in (MAEP-0) has not been studied before to the best of our understanding. One approach would be to pose it as an integer optimization problem. One needs to impose constraints for traversing every arc once and for sub-tour elimination. Indeed, such a formulation is the basis of the quantum approach in Section 3.2.1.

An alternative is to use canonical problems to construct approximation algorithms for MAEP. A natural candidate is the Traveling Salesman Problem (TSP) because it can be easily converted to Eulerian tour problem by doubling edges. Approximation algorithms for mTSP are available in [11, 12]. TSP algorithms are well supplemented with solvers such as `concorde` [13], based on established approximations [14, 15]. An another canonical problem which we shall explore is the *Chinese postman problem* [16, 17]. Besides, we have considered partitioning based algorithms for a variety of multi agent resource allocation in our previous work [18, 9]. Our objective is to develop a strong baseline to compare quantum algorithms.

3.2 Quantum Solution Approach

3.2.1 Quantum Approximate Optimization Algorithm (QAOA) Approach

Consider the following QAOA approach to solving MAEP-0. One needs to find a bitstring which corresponds to the optimal sequence of links. If n is the number of links, then we use n^2m binary variables. We divide the bitstring x into n sections of nm variables; let these sections be denoted as $x_1, \ldots, x_n$. Let each of these sections be further divided into m subsections of n variables each. Let these subsections of x_i be denoted as $x_{i,1}, \ldots, x_{i,m}$. These subsections correspond to the flight segments assigned to the agents for the i-th leg of their daily itinerary. In particular, at most one of the bits in $x_{i,j}$ is 1, for every $j \in \{1, \ldots, m\}$; all are zero if the itinerary of the i-th agent contains less than j segments. In summary, $x_{i,j,k} = 1$ if link k is the i-th flight segment of agent j, and equal to zero otherwise. A sample bitstring for $n = 4$ and $m = 2$ is:

$$x = \overbrace{\underbrace{1,0,0,0}_{x_{1,1}},\underbrace{0,1,0,0}_{x_{1,2}}}^{x_1},\overbrace{\underbrace{0,0,1,0}_{x_{2,1}},\underbrace{0,0,0,1}_{x_{2,2}}}^{x_2},\overbrace{\underbrace{0,0,0,0}_{x_{3,1}},\underbrace{0,0,0,0}_{x_{3,2}}}^{x_3},\overbrace{\underbrace{0,0,0,0}_{x_{4,1}},\underbrace{0,0,0,0}_{x_{4,2}}}^{x_4}$$

According to this bitstring, aircraft 1's schedule is to service segment 1 followed by 3, and aircraft 2's schedule is to service segment 2 followed by 4. The bitstrings x have to satisfy constraints which will be formalized later when we construct the cost Hamiltonian.

The general steps in QAOA are: (i) Prepare the initial state; (ii) Apply the cost Hamiltonian; (iii) Apply the driver Hamiltonian; (iv) Exponentiate and parameterize the resulting product by γ's and β's; and (v) Use classical algorithms to optimize over β and γ.

Let the driver Hamiltonian be $\hat{\mathcal{H}}_D = \sum_i X_i$, where X_i denotes the Pauli operator acting on the i-th qubit. The ground state of the driver Hamiltonian is $|H_D^0\rangle = \frac{1}{2^{n^2m}} \sum_{z \in \{0,1\}^{n^2m}} |z\rangle$. The following four terms constitute the cost Hamiltonian, i.e., $\hat{\mathcal{H}}_C = \hat{\mathcal{H}}_1 + \hat{\mathcal{H}}_2 + \hat{\mathcal{H}}_3 + \hat{\mathcal{H}}_4$: the first three characterize constraints, and the fourth defines the objective:

1. Each link should be visited exactly once: $\hat{\mathcal{H}}_1 = \sum_{k=1}^{n} \left(1 - \sum_{i=1}^{n} \sum_{j=1}^{m} x_{i,j,k}\right)^2$

2. Every aircraft should serve at most one link at any time:
   $\hat{\mathcal{H}}_2 = \sum_{i=1}^{n} \sum_{j=1}^{m} \sum_{k=1}^{n} x_{i,j,k} \left(1 - x_{i,j,k}\right)^2$
3. One link can be visited after the other by an aircraft, only if they are adjacent to each other:
   $\hat{\mathcal{H}}_3 = \sum_{j=1}^{m} \sum_{i=1}^{n-1} \left(\sum_{k_1 : x_{i,j,k_1}=1} \sum_{k_2 : x_{i+1,j,k_2}=1} \mathbf{1}_{(k_1 \text{ incident to } k_2)} \right)$
4. Minimize the total cost: $\hat{\mathcal{H}}_4 = \max_{j \in \{1,\ldots,m\}} \sum_{i=1}^{n-1} \sum_{k=1}^{n} c_j(k)$, where $c_j(k)$ is the cost to serve link k by the j-th aircraft.

The cornerstone of our analysis will be the *folk theorem* of QAOA, i.e., as the size of the parameters β and γ grow larger, does the output of QAOA converge to optimal value for MAEP? In terms of implementation on a quantum computer, if the Hamiltonians are quadratic, then it is known that they can be realized as sums of Pauli matrices, e.g., see [2, 3]. The non-quadratic nature of the Hamiltonians in our problem can be handled by an embedding-based relaxation method, e.g., see [19]. We shall investigate the feasibility and tightness of such relaxations for MAEP-1 and MAEP-2. We shall also explore generalization to parameterized families of unitaries which contain the specific Hamiltonians described above, along the lines of [20]. Finally, specifically to model temporal constraints in MAEP-2, we shall explore equivalent mixed integer program formulations, e.g., as illustrated in [21] in the context of vehicle routing problems. While the literature on quantum algorithms for mixed programs is scarce, relaxations to continuous programs [22] is a natural starting point.

3.2.2 Generalized QAOA: Partitioning For Online-Offline Computation

Recall that implementation of QAOA is in fact a bilevel optimization approach (Figure 2), with quantum computing constituting the inner loop, and the outer loop corresponding to optimization of parameters β and γ. One can generalize such a *hybrid* approach even further. We propose to pursue such a generalization to enable fast *online* adaptation to dynamic situations or even an abrupt change. This would be necessitated, e.g., when an airline faces a disruption such as the one caused by the pandemic. Our approach here is inspired by the one in [4] for mixed-integer programs. The key idea is to partition the decision variables into two sets: one will be optimized in an offline manner and the other in an online manner. The optimization of offline variables is accomplished by creating a map from problem parameters to optimal values. Such a map can be created through exhaustive search or by training a neural network on sample scenarios. In real-time, given problem parameters, the optimal value of offline variables is obtained from the map and the online variables are solved conventionally. We propose to add such a partitioning layer to QAOA (see Figure 2) and study it in the context of MAEP-1 and MAEP-2. While it was somewhat natural to partition among integer and continuous variables in [4], in our context, one needs to carefully

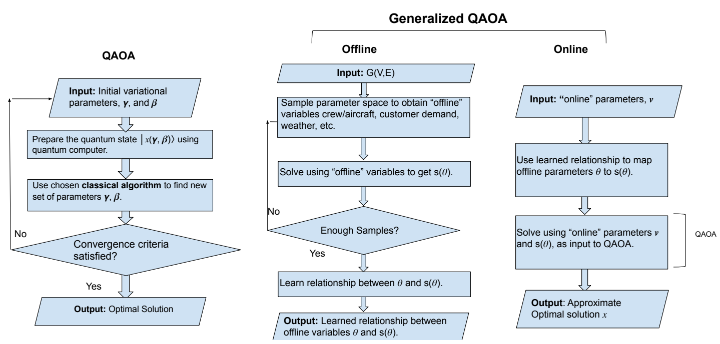

Figure 2: Generalization of QAOA.

tradeoff computational speedup with loss in accuracy due to decoupling of online and offline variables.

3.2.3 Quantum Graph Optimization Approach (QGOA)

While the near term quantum computing hardware is anticipated to conducive for QAOA implementation, in the longer term as more architectures become available, it would be natural to also look at other quantum computing paradigms which might show better speedups. With this perspective in mind, we plan to study algorithms based on quantum queries, especially for graph problems, e.g., see [5]. This approach looks for a combination of data structure (e.g., adjacency matrix vs. array) for specifying problem parameters and a search algorithm [23, 24, 25] to give speedups, possibly in comparison with classical approaches. Such an approach has proven to be successful on several canonical problems, including the minimum spanning tree. Given the lack of efficient classical algorithms for MAEP problems (which we also propose to address – see Section 3.1), we believe that QGOA is a promising approach which has been completely ignored in airline scheduling problems to the best of our knowledge.

3.3 Simulation Implementation

The implementation for solving the airline scheduling problem, and hence the multi-agent Eulerian path problem, will involve both classical and quantum approaches. Software implementation will be generated with the assistance of tools freely available from IBM: Qiskit for simulation, and IBM OpenQuantum for quantum computation.

3.3.1 Quantum Computing Implementation

In both simulation using Qiskit and quantum computing using Open Quantum, we are allotted a fixed number of qubits N, which will determine the maximum size of the problem we will solve. Here we give an estimate of how the number of qubits N depends on our problem parameters in section 3.1.

QAOA involves Adiabatic Quantum Computation, which in practice is achieved by expressing the cost Hamiltonian as an Ising Hamiltonian [26]. Note that our cost Hamiltonian $\hat{\mathcal{H}}_c$ in section 3.1 is not an Ising Hamiltonian because it is not inherently quadratic. Quadratic terms are easily handled by the Ising model, but the cubic terms must be transformed or approximated, using for example minor-embedding [19]. But generally, any order interaction can be supported by the Ising model [27]. We will use the Ising Hamiltonian for a Eulerian path from [28] as a starting point, and eventually obtain a cost Hamiltonian of the following form: $\hat{\mathcal{H}}_c(\sigma_1, ..., \sigma_N) = -\sum_{i<j} J_{ij}\sigma_i^z\sigma_j^z - \sum_{i=1}^{N} h_i\sigma_i^z$, where σ_i^z is a Pauli-z operator. The cost Hamiltonian $\hat{\mathcal{H}}_c$ expressed in the form above (Ising Hamiltonian), has 2^N possible assignments for the aircraft scheduling problem. Therefore since we need n^2m binary variables to represent 2^{n^2m} possible values for $\hat{\mathcal{H}}_c$, we compute the number of required qubits to be $N = n^2m$. This parameter can be specified by passing inputs into the qasmSimulator - a module in Qiskit [6].

3.3.2 Circuit Depth

In QAOA, the expected value of the objective improves monotonically with increasing the dimension p of the variational parameters [2], so we are incentivized to make p as large as possible. However, in implementation, we are limited by the depth of the quantum circuit.

The depth of the circuit is $p*(m+1)$ and grows linearly with the number of clauses m in the MaxSAT problem [2]; With 4 clauses, as described in section 3.1, we require the depth of the circuit would scale in the worst case as $p*(4+1) = 5*p$. Due to its stochastic nature, in measuring success of our QAOA implementation, of particular interest is the expectation of the objective function, denoted $F_p(\gamma, \beta)$, and how it changes with the number of repetitions. From [2], it is known that for MaxCut, which is a special case of MaxSAT, an outcome of F_p–1 can be obtained with probability $1 - \frac{1}{m}$ from $m * \log m$ iterations. When formed as a MaxSAT problem, our aircraft assignment problem has 4 clauses, which we approximate will lead to the same outcome, F_p–1 with probability $\frac{3}{4}$ after only 6 runs of QAOA.

3.3.3 Classical to Quantum: Expected Computational Results

Algorithms for finding an Eulerian directed path in a digraph are generally derived from the Splitting Algorithm, which is in the worst case $O(|V| * |E|)$ [29]. Since our MAEP problem is a variation on the Eulerian path problem, in theory we expect a similar runtime on a classical computer, perhaps differing by a constant factor to find multiple paths, using multiple agents. More empirically speaking, the time required to solve the Eulerian Cycle problem on a quantum computer has been experimentally shown to result in a speed-up of two orders of magnitude: for 40 edges, a classical computer requires 10^9 operations, whereas a quantum algorithm required $4 * 10^7$ [30].

3.3.4 Choice of the Driver Hamiltonian

As mentioned previously in Section 3.1, a desirable property of QAOA is that as the number of parameters p increases, the expected value of the cost Hamiltonian $\hat{\mathcal{H}}_C$ approaches the true optimal value of the objective function. However, this property is only maintained by a careful choice of the driver Hamiltonian, $\hat{\mathcal{H}}_D$, since this determines the ground state in which we start the adiabatic evolution. This is a property that is not offered in some algorithms such as the quantum adiabatic algorithm (QAA) [2]. We will research whether we may take advantage of this property by appropriately choosing $\hat{\mathcal{H}}_D$ for both MAEP-1 and MAEP-2.

4 Expected Significance & Roadmap to Industry Adoption

The project will provide a principled approach to speed up solution to large scale airline problems by combining strengths of classical and quantum computing with problem structure. The insights from analysis and simulation will help to critically evaluate rules of thumb for dynamic matching between crew and aircrafts that are commonly used in practice. The computational gains will benefit baseline operation plan and quick response to disruptions.

Current approaches to aircraft/crew routing/scheduling typically use generic heuristics from integer programming, which do not provide universal guarantees on the approximation quality and do not sufficiently exploit problem structure. In contrast, our approach centers around specific formulations well-suited to practical setups. Furthermore, our work will substantially expand frontiers in the nascent field of quantum computing for airline problems.

A substantial increase in air travel is expected over the next couple of decades, which will in turn increase operational complexity. The same period is expected to coincide with breakthroughs in quantum computing technology. Therefore, naturally quantum computing is poised to play a crucial role is solving airlines' complex business problems in the future [31]. Indeed, quantum advantage will prove to be a game-changer for airlines to optimize operations and improve customer experience. This project is intended to provide credence to such a vision and to inform concrete directions for further research in this area.

Technology Readiness: Our research plan spans TR1-7. The validation part (TR8-9) will require access to bigger scale quantum computing resources. Towards that purpose, we plan to pursue collaboration with USC's Center for Quantum Information Science & Technology (CQIST) following this PWICE project.

5 Plan of Work

Q1: 7/1-9/30 | Q2: 10/1-12/31 | Q3: 1/1-3/31 | Q4: 4/1-6/30

Year 1: 7/1/2022-6/30/2023

Milestone	Outcomes	Lead	Q1	Q2	Q3	Q4
1	Formalization of MAEP-1 and MAEP-2	Hur	■			
2	Literature review on classical algorithms for MAEP-1 and MAEP-2	Hur	■	■		
3	Literature review on quantum algorithms	Savla	■	■		
4	Classical approximation algorithm for MAEP-1	Hur		■	■	
5	QAOA algorithm for MAEP-1	Savla		■	■	
6	Classical simulation comparison of classical and QAOA approaches for MAEP-1 using Qiskit	Savla			■	■

Year 2: 7/1/2023-6/30/2024

Milestone	Outcomes	Lead	Q1	Q2	Q3	Q4
1	Classical approximation algorithm for MAEP-2	Hur	■	■		
2	QAOA algorithm for MAEP-2	Savla	■	■		
3	Classical simulation comparison of classical and QAOA approaches for MAEP-2	Hur		■	■	
4	QGOA algorithm for MAEP-1 or MAEP-2	Savla		■	■	
5	Classical simulation of classical algorithm vs. QAOA of MAEP-1 or -2 using IBM Open Quantum	Hur			■	■
6	Generalized QAOA (Section 3.2.2) for MAEP-1 or MAEP-2	Savla			■	■

6 Team Member Contributions

Youngbum Hur (Inha) worked on "quantum computing for airline optimization problems" at Sabre research team and won the "Best Innovation Award" at 2018 AGIFORS. He works on discrete optimization and data-driven approaches for large-scale problems in manufacturing, transportation & logistics. He will be supported by one postdoc his contribution to this project will be in providing airline related MAEP problems and their formulations, and classical approximation algorithm and QAOA implementation. **Ketan Savla** (USC) is a co-founder & chief science advisor to an urban mobility startup. He has extensive experience in solutions to large scale problems under explicit practical constraints from transportation & robotics. He will be supported by one PhD student. His contribution to this project will be in designing constant factor approximation classical algorithms, QAOA and QGOA algorithms for variants of MAEP, and their simulation implementation. The Inha and USC PIs will meet with their respective research teams on a weekly basis. The entire team will meet (virtually) biweekly, leading to joint publications and algorithm code.

References

[1] Muhammet Deveci and Nihan Çetin Demirel. A survey of the literature on airline crew scheduling. *Engineering Applications of Artificial Intelligence*, 74:54–69, 2018.

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

[3] Pontus Vikstål, Mattias Grönkvist, Marika Svensson, Martin Andersson, Göran Johansson, and Giulia Ferrini. Applying the quantum approximate optimization algorithm to the tail-assignment problem. *Physical Review Applied*, 14(3):034009, 2020.

[4] Dimitris Bertsimas and Bartolomeo Stellato. Online mixed-integer optimization in milliseconds. *arXiv preprint arXiv:1907.02206*, 2019.

[5] Christoph Dürr, Mark Heiligman, Peter HOyer, and Mehdi Mhalla. Quantum query complexity of some graph problems. *SIAM Journal on Computing*, 35(6):1310–1328, 2006.

[6] IBM Research. *QasmSimulator - Qiskit 0.34.2 documentation*. IBM Qiskit Community, 2018.

[7] IBM Research. *IBM Open Quantum*, 2022.

[8] IBM Research. *Introduction to Quantum Error Correction using Repetition Codes*. QisKit Community, 2021.

[9] F. Bullo, E. Frazzoli, M. Pavone, K. Savla, and S. L. Smith. Dynamic vehicle routing for robotic systems. *Proceeedings of the IEEE*, 99(9):1482–1504, 2011. Invited paper.

[10] Pavel A Pevzner, Haixu Tang, and Michael S Waterman. An eulerian path approach to dna fragment assembly. *Proceedings of the national academy of sciences*, 98(17):9748–9753, 2001.

[11] Alan M Frieze. An extension of christofides heuristic to the k-person travelling salesman problem. *Discrete Applied Mathematics*, 6(1):79–83, 1983.

[12] Sivakumar Rathinam, Raja Sengupta, and Swaroop Darbha. A resource allocation algorithm for multivehicle systems with nonholonomic constraints. *IEEE Transactions on Automation Science and Engineering*, 4(1):98–104, 2007.

[13] Concorde tsp solver. `https://www.math.uwaterloo.ca/tsp/concorde.html`.

[14] Nicos Christofides. Worst-case analysis of a new heuristic for the travelling salesman problem. Technical report, Carnegie-Mellon Univ Pittsburgh Pa Management Sciences Research Group, 1976.

[15] Keld Helsgaun. An effective implementation of the lin–kernighan traveling salesman heuristic. *European journal of operational research*, 126(1):106–130, 2000.

[16] Wen Lea Pearn. Solvable cases of the k-person chinese postman problem. *Operations Research Letters*, 16(4):241–244, 1994.

[17] Dino Ahr and Gerhard Reinelt. New heuristics and lower bounds for the min-max k-chinese postman problem. In *European symposium on algorithms*, pages 64–74. Springer, 2002.

[18] M. Pavone, K. Savla, and E. Frazzoli. Sharing the load. *IEEE Robotics and Automation Magazine*, 16(2):52–61, 2009.

[19] Vicky Choi. Minor-embedding in adiabatic quantum computation: I. the parameter setting problem. April 2008.

[20] Stuart Hadfield, Zhihui Wang, Bryan O'gorman, Eleanor G Rieffel, Davide Venturelli, and Rupak Biswas. From the quantum approximate optimization algorithm to a quantum alternating operator ansatz. *Algorithms*, 12(2):34, 2019.

[21] Sertac Karaman and Emilio Frazzoli. Linear temporal logic vehicle routing with applications to multi-uav mission planning. *International Journal of Robust and Nonlinear Control*, 21(12):1372–1395, 2011.

[22] Fernando GSL Brandao and Krysta M Svore. Quantum speed-ups for solving semidefinite programs. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 415–426. IEEE, 2017.

[23] LK Grover. A fast mechanical algorithm for database search. In *Proceedings of the 28 Annual ACM Symposium on the Theory of Computing (Philadelphia, Pennsylvania, 22-24 May 1996)*, page 212, 1996.

[24] Gilles Brassard and Peter Hoyer. An exact quantum polynomial-time algorithm for simon's problem. In *Proceedings of the Fifth Israeli Symposium on Theory of Computing and Systems*, pages 12–23. IEEE, 1997.

[25] Gilles Brassard, Peter Hoyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. *Contemporary Mathematics*, 305:53–74, 2002.

[26] Sergio Boixo, Tameem Albash, Federico M. Spedalieri, Nicholas Chancellor, and Daniel A. Lidar. Experimental signature of programmable quantum annealing. *Nature Communications*, 4(1):2067, Jun 2013.

[27] Google. *QAOA Ising Tutorial*. Google Colaboratory, Mountainview, CA, 2018.

[28] Andrew Lucas. Ising formulations of many np problems. *Frontiers in Physics*, 2, 2014.

[29] Herbert. Fleischner. *Eulerian graphs and related topics. Part 1, Volume 2*. Annals of discrete mathematics ; 50. North-Holland, Amsterdam ;, 1991. Includes bibliographical references and indexes.

[30] Massoud Amini. Quantum algorithms of decision and planning problems. *Mathematics Subject Classification*, 1:5–85, 01 2009.

[31] Imed Othmani. Exploring quantum computing use cases for airlines. `https://www.ibm.com/thought-leadership/institute-business-value/report/quantum-airlines`, March 2020.