

Article

Variational Quantum Algorithm Applied to Collision Avoidance of Unmanned Aerial Vehicles

Zhaolong Huang¹, Qiting Li^{2,*} , Junling Zhao¹ and Meimei Song^{1,*}

¹ College of Science, Tianjin University of Technology, Tianjin 300384, China

² R & D Department, China Academy of Launch Vehicle Technology, Beijing 100076, China

* Correspondence: jeetinlee@gmail.com (Q.L.); songmeimei@tjut.edu.cn (M.S.)

Abstract: Mission planning for multiple unmanned aerial vehicles (UAVs) is a complex problem that is expected to be solved by quantum computing. With the increasing application of UAVs, the demand for efficient conflict management strategies to ensure airspace safety continues to increase. In the era of noisy intermediate-scale quantum (NISQ) devices, variational quantum algorithms (VQA) for optimizing parameterized quantum circuits with the help of classical optimizers are currently one of the most promising strategies to gain quantum advantage. In this paper, we propose a mathematical model for the UAV collision avoidance problem that maps the collision avoidance problem to a quadratic unconstrained binary optimization (QUBO) problem. The problem is formulated as an Ising Hamiltonian, then the ground state is solved using two kinds of VQAs: the variational quantum eigensolver (VQE) and the quantum approximate optimization algorithm (QAOA). We select conditional value-at-risk (CVaR) to further promote the performance of our model. Four examples are given to validate that with our method the probability of obtaining a feasible solution can exceed 90% based on appropriate parameters, and our method can enhance the efficiency of a UAVs' collision avoidance model.

Keywords: collision avoidance of UAVs; variational quantum algorithms; variational quantum eigensolver; quantum approximate optimization algorithm; conditional value-at-risk



Citation: Huang, Z.; Li, Q.; Zhao, J.; Song, M. Variational Quantum Algorithm Applied to Collision Avoidance of Unmanned Aerial Vehicles. *Entropy* **2022**, *24*, 1685. <https://doi.org/10.3390/e24111685>

Academic Editor: Giuliano Benenti

Received: 30 October 2022

Accepted: 12 November 2022

Published: 18 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The emergence of UAVs has dramatically changed the way people use transportation [1], such as aerial logistics networks and short- and medium-range manned transportation at low altitude. UAVs have the characteristics of a small safety risk factor and low cost. Therefore, in the foreseeable future, the frequency of UAV use will rise, and the workload of low-altitude air traffic flow management (ATFM) will also increase accordingly. Deploying UAVs on a large scale faces an issue of collision avoidance problem of UAVs. Therefore, it is necessary to build an algorithm for unmanned traffic management (UTM) [2]. We need to determine the UAVs' collision avoidance actions, including takeoff delays and local maneuvers, that the aircraft needs to implement in real time according to the potential conflicts of different aircraft routes, and avoid conflicts while ensuring a low cost of collision avoidance. Mathematically, this problem is an optimization problem to avoid collisions at minimum cost. With the expansion of the problem scale and the increase of constraints, the required computing resources also increase exponentially, which leads to increasing difficulty of solving optimization problems.

As quantum computers have continued to advance in recent decades, researchers have begun to use quantum computers to find solutions to optimization problems [3–5]. Quantum computers have the capability of parallel computing and considerable computing advantages over classical computers. In some specific tasks, they have even surpassed the current most advanced classical computers [6–8]. Current quantum computers are noisy intermediate-scale quantum (NISQ) computers [9]. Although the most advanced quantum

computers have reached fifty to hundreds of qubits, they are still unable to achieve fault tolerance, hence fault-tolerant quantum computers may not be available in the short term. VQA combines the advantages of quantum and classical computing by using parametric quantum circuits running on a quantum computer followed by parametric optimization on a classical optimizer [10]. Utilizing classical computers, VQA, not only compensates for the shortage of quantum bits and pushes the complexity to classical computers, but running only short-depth quantum circuits is an extremely effective strategy to reduce errors in NISQ devices.

Given that VQA, with promising properties, can support a form of quantum advantage [11,12], and considerable progress have been made in its experimental realization [13], we select popular VQA, such as the variational quantum eigensolver (VQE) [14] and the quantum approximate optimization algorithm (QAOA) [5], to solve our collision avoidance problem. Furthermore, in order to enhance the efficiency of VQA, we also introduce the recently proposed conditional value-at-risk (CVaR) as an aggregation function to optimize the original cost function and compare the performance with the original VQA [4].

When solving optimization problems with quantum computers, the optimization problem is often formulated as a quadratic unconstrained binary optimization (QUBO) problem [15]. In the transportation domain, researchers have used quantum computing to solve air traffic management problems. A kind of air traffic management (ATM) problem has been converted to a QUBO problem and handled by a quantum computer [3]. Additionally, quantum computing is also being used for railway conflict management problems [16].

In the field of public transportation, UAVs are gaining attention as a promising means of transportation, and they have become one of the main available options for the transportation of medical supplies [17]. Internationally, there are already many mature and well established applications for drone transport, such as the East African medical drone transport [18], Amazon's PrimeAir drone delivery service, etc. In 2021, Sumitomo Japan developed thousands of flight paths for drones through quantum computing for Urban Air Mobility [19]. In order to maintain the efficiency and safety of the both goods and people, the National Aeronautics and Space Administration (NASA) proposed the concept of Urban Air Mobility (UAM) in 2017 [20].

UAV transportation missions require huge numbers of UAVs to collaborate, which involves the route planning of multiple UAVs that include collision avoidance, task allocation, and so on. In this paper, we study the collision avoidance problem of UAVs based on VQA. We consider not only spatio-temporal constraints, but also the necessary factors such as preparation time, early takeoff time, and heterogeneity among multiple UAVs that should be accounted in practical missions. Inspired by the QUBO method in [3], we mathematically model the UAV collision avoidance problem in Section 2 through considering the collision avoidance by delaying the takeoff time of the aircraft and transforming the problem into a QUBO problem after discretization. Following that, we discuss the Ising model for the QUBO problem and related concepts of the variational quantum algorithm in Section 3. In Section 4, we present the test results of the four examples. Finally, we present a summary and outlook of the article.

Based on the particulars of the actual UAV flight, we considered the following conditions:

- Spatio-temporal conditions. For the objective requirement of the aircraft to ensure safety of flight, the distance between any two aircraft at the same point in time must be greater than a threshold value. From the standpoint of flight tasks, this requires us to adjust the trajectory of the aircraft or change the time of arrival at the conflict point when a conflict occurs. If we put it in mathematical perspective, it means that we must ensure that, for at least one of the two, the spatial function A and the temporal function T , should be bigger than 0.
- Preparation condition. Because of the special features of UAVs, the aircraft require preparation before takeoff. Nevertheless, the preparation time needed before takeoff

is different for various types of aircraft, so we set a non-negative extra time for each aircraft, which is named pre-takeoff preparation time.

- Early takeoff condition. Unlike civil flights carrying people, in the mission of the UAVs we allow aircraft to take off earlier than scheduled departure time in order to avoid conflicts. This means that the departure delay time of the aircraft can not only be positive but also be negative.
- Heterogeneity condition. A UAV formation has different types and different models of aircraft that are required to perform different missions. For the priority of the aircraft resulting from the importance of missions to be performed or the urgency of tasks to be executed, we assign corresponding weights to each aircraft.

2. UAV Collision Avoidance Theory

The collision avoidance problem of UAVs is a complex problem and an important part of route planning. The input of the problem is a series of aircraft trajectories. For the purpose of saving flight cost, we tend to choose the optimal flight trajectory that has the lowest flight cost and minimum flight time when planning the route for the aircraft. This leads to the possibility that the optimal flight paths chosen by different aircraft will overlap with one another, thus increasing the probability of aircraft collisions. In this paper, assuming that the trajectories are known, we focus on the collision elimination by adjusting the takeoff time.

In the collision avoidance problem, the most common solution strategy is to change course or speed, but this also inevitably increases the cost of flight. However, the easiest way to de-conflict trajectory conflicts while taking into account flight costs is to delay the departure of aircraft having conflicting routes. When the delay time is short, there is hardly any negative impact on the completion of most tasks. This is the strategy we will utilize and analyze in this article.

2.1. Problem Description

In this research, it is assumed that delaying the takeoff time of a certain aircraft will not introduce new conflicts, and the input trajectory is $\mathbf{A}_i = (\mathbf{A}_{i,t_i})$, where $\mathcal{A} = \{i | i \in [1, F]\}$ represents the set of aircraft, $\mathbf{A}_{i,t}$ represents the position of the aircraft i at time t (including its longitude, latitude, and altitude), $t_i \in [\tau_{i,0}, \tau_{i,1}]$ is the flight time of aircraft i , and $\tau_{i,0}$ and $\tau_{i,1}$ are the takeoff time and landing time of i , respectively.

Let the delay when the aircraft i arrives at the point \mathbf{A}_{i,t_i} be D_{i,t_i} and the maximum and minimum delay allowed is, respectively, D_{i,t_i}^{\max} and D_{i,t_i}^{\min} , then the total delay of all aircraft is

$$D = \sum_{i=1}^F D_{i,\tau_{i,1}}. \tag{1}$$

The waypoint $\mathbf{A}_{i,t_i}, \mathbf{A}_{j,t_j}$ of a pair of aircraft (i, j) will not conflict if and only if their space and time functions are greater than their respective separation criteria Δ_a and Δ_t , where Δ_a and Δ_t are non-negative real numbers. For example, the position separation criteria Δ_a can be set to 10 m, and the time separation Δ_t can be set to 2 s. In the research on the collision avoidance of UAVs, the following conditions should be met:

$$\max\{A, T\} > 0, \tag{2}$$

where

$$A = \left\| \mathbf{A}_{i,t_i} - \mathbf{A}_{j,t_j} \right\| - \Delta_a, \tag{3}$$

$$T = \left\| (t_i + D_{i,t_i}) - (t_j + D_{j,t_j}) \right\| - \Delta_t. \tag{4}$$

Consider the values of A and T , if (i, j) exists a collision, then there must be $\mathbf{A}_{i,t_i}, \mathbf{A}_{j,t_j}$ such that $A < 0$ and $T < 0$. By Equation (4) we can find that, if $T < 0$, then

$$|t_i - t_j| - |D_{j,t_j} - D_{i,t_i}| \leq |(t_i - t_j) - (D_{j,t_j} - D_{i,t_i})| < \Delta t, \tag{5}$$

which implies that

$$|D_{j,t_j} - D_{i,t_i}| + \Delta t > |t_i - t_j|. \tag{6}$$

It can be seen that $(D_{i,t_i}, D_{j,t_j}) \in [D_{i,t_i}^{\min}, D_{i,t_i}^{\max}] \times [D_{j,t_j}^{\min}, D_{j,t_j}^{\max}]$, i.e.,

$$\begin{aligned} D_{i,t_i}^{\min} &\leq D_{i,t_i} \leq D_{i,t_i}^{\max}, \\ D_{j,t_j}^{\min} &\leq D_{j,t_j} \leq D_{j,t_j}^{\max}. \end{aligned} \tag{7}$$

It follows from Equation (7) that

$$D_{i,t_i}^{\min} - D_{j,t_j}^{\max} \leq D_{i,t_i} - D_{j,t_j} \leq D_{i,t_i}^{\max} - D_{j,t_j}^{\min}. \tag{8}$$

Combining Equations (6) and (8) yields

$$\max\{|D_{i,t_i}^{\max} - D_{j,t_j}^{\min}|, |D_{j,t_j}^{\max} - D_{i,t_i}^{\min}|\} + \Delta t > |t_i - t_j|, \tag{9}$$

and it is clear that there exists (D_{i,t_i}, D_{j,t_j}) making $T < 0$, so the trajectories of (i, j) have the risk of conflict. Assuming that the closest conflict point of the trajectory distance of (i, j) is M , then there must be other potential conflict points near M .

As shown in Figure 1, in the adjacent area C of M , we have $A < 0$. Thus, if A_{i,t'_i} conflicts with A_{j,t'_j} , and A_{i,t''_i} also conflicts with A_{j,t''_j} , then for any $t_i \in [\min\{t'_i, t''_i\}, \max\{t'_i, t''_i\}]$, there exists $t_j \in [\min\{t'_j, t''_j\}, \max\{t'_j, t''_j\}]$ for which A_{i,t_i} conflicts with A_{j,t_j} . These conflicts can be called conflicts derived from point M , and we can express the whole of the conflicts derived from point M as C_m . Similarly, the set of all aircraft collisions can be expressed as the set of mutually disjoint C_m ,

$$C = \bigcup C_m, \tag{10}$$

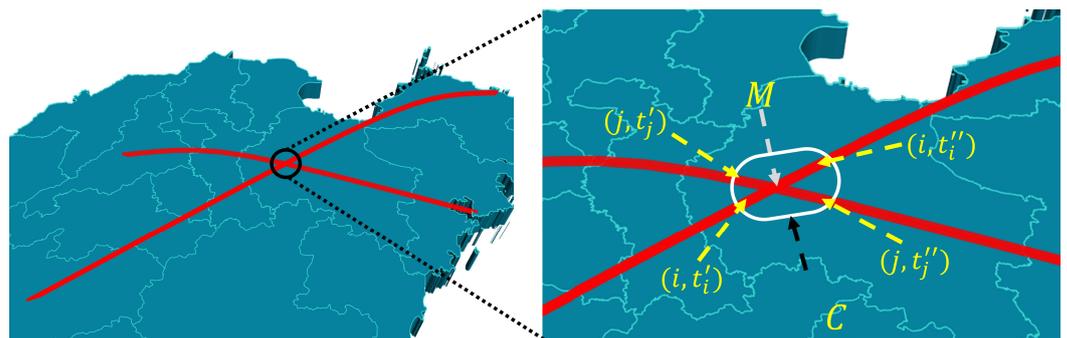


Figure 1. Schematic diagram of a pair of conflicting routes.

We set the aircraft related to the conflict m as $\mathcal{R}_m = \{i \mid i \in C_m\}$, then the set of conflicts related to the aircraft i is $M_i = \{m \mid i \in \mathcal{R}_m\}$. Let $d_{i,m}$ be the scheduling delay of aircraft i , which is the delay incurred to prevent the collision m ; $d_{i,p}$ is the pre-takeoff preparation time of aircraft i . This means that the total delay before aircraft i encounters collision m is

$$\begin{aligned} D_{i,m} &= d_{i,p} + \sum_{m' \in M_{i,m}} d_{i,m'}, \\ M_{i,m} &= \{m' < m \mid m' \in M_i\}. \end{aligned} \tag{11}$$

Then, for the conflict m between aircraft i and j , the total time pair satisfying the condition $T < 0$ can be written as

$$\mathcal{T}_m = \{(t_i, t_j) \mid \{(i, t_i), (j, t_j)\} \in C_m, i < j\}, \tag{12}$$

where t_i and t_j denote times as non-negative real numbers. We can obtain a range \mathcal{B}_m from \mathcal{T}_m ,

$$\begin{aligned} \mathcal{B}_m &= \bigcup_{(t_i, t_j) \in \mathcal{T}_m} (-\Delta_t + t_j - t_i, \Delta_t + t_j - t_i), \\ &= \left[1 - \Delta_t + \min_{(t_i, t_j) \in \mathcal{T}_m} (t_j - t_i), \Delta_t - 1 + \max_{(t_i, t_j) \in \mathcal{T}_m} (t_j - t_i) \right]. \end{aligned} \tag{13}$$

If it is satisfied that $D_m = D_{i,m} - D_{j,m} \notin \mathcal{B}_m$, which is equivalent to potential conflicts being avoided. Therefore, we can express the total delay as

$$D = \sum_{i=1}^F \left(d_{i,p} + \sum_{m \in M_i} d_{i,m} \right). \tag{14}$$

In order to reduce unnecessary calculation costs, we can remove the aircraft with no conflicts when calculating the delay time, which means we only need to re-calculate the aircraft $i \in [1, N_c] (N_c \leq F)$ with conflict risk. We want to minimize the value of D while eliminating collisions through optimization. At the same time, we only consider the conflict avoidance by delaying the departure time of the aircraft, which means that for aircraft i , we can use d_i to denote $\sum_{m \in M_i} d_{i,m}$, namely, the scheduling delay time of aircraft i . Thus, the total delay can be simplified as

$$D = \sum_{i=1}^{N_c} D_i, \tag{15}$$

where $D_i = d_{i,p} + d_i$ is the delay sum of aircraft i . The configuration space of the problem is $\mathcal{D} = \{D_i \mid i \in [1, N_c]\}$. Taking into account the importance of missions or the urgency of tasks, we assign weights to the delays of each aircraft, and the above formula can be transformed into

$$D = \sum_{i=1}^{N_c} f(i) \cdot D_i. \tag{16}$$

2.2. Mapping Problem to QUBO

We first encode the configuration space \mathcal{D} as a series of binary variables, and then solve the collision avoidance problem with quantum computing. To achieve this, we must first discretize the delay time allowed by the aircraft. Let Δ_d be the discretization step size of the delay. Then $d_i^{\max} = N_i^{\max} \cdot \Delta_d$ is the maximum scheduling delay time allowed by aircraft i , so $d_i \in \{\Delta_d l \mid l \in [0, N_i^{\max}]\}$. At the same time, in some cases, we can also avoid conflicts by advancing the departure time of the aircraft, that is, the minimum scheduling delay time $d_i^{\min} = N_i^{\min} \cdot \Delta_d < 0$, therefore $d_i \in \{l\Delta_d \mid l \in [N_i^{\min}, N_i^{\max}]\}$. Similarly, $d_{i,p}$ can also be expressed as $d_{i,p} = p_i \cdot \Delta_d$, then $D_i = \ell\Delta_d, \ell \in [p_i + N_i^{\min}, p_i + N_i^{\max}]$. As the configuration space increases, the number of qubits required for encoding will also increase, so Δ_d has a great impact on the quality of the solution. Next we describe how to map the collision avoidance problem to the QUBO problem.

We need to encode D_i as a binary variable after discretizing the configuration space. In this case, we introduce a decision variable

$$x_{i,\ell} = \begin{cases} 1, & D_i = \ell\Delta_d \\ 0, & \text{otherwise} \end{cases}. \tag{17}$$

If the aircraft takes off with delay D_i , then $x_{i,\ell}$ takes the value 1; otherwise $x_{i,\ell}$ takes 0. Thus, the weighted sum of the total delays is

$$f = \sum_{i=1}^{N_c} D_i = \Delta_d \sum_{i=1}^{N_c} \sum_{\ell=p_i+N_i^{\min}}^{p_i+N_i^{\max}} f(i) \cdot \ell x_{i,\ell}. \tag{18}$$

To transform the problem into an unconstrained problem, we define a function with penalty. We have discussed above that if (13) holds, potential conflicts can be avoided, which is also a hard constraint for collision avoidance:

$$\mathcal{P}_{con} = p_{con} \sum_m \sum_{\substack{\ell, \ell' | \Delta_d(\ell - \ell') \in \mathcal{B}_m \\ i, j \in \mathcal{R}_m | i < j}} x_{i,\ell} \cdot x_{j,\ell'}, \tag{19}$$

where p_{con} is penalized for a sufficiently large positive constant. Because each aircraft can only choose one total delay, namely

$$\sum_{\ell=p_i+N_i^{\min}}^{p_i+N_i^{\max}} x_{i,\ell} = 1, i \in [1, N_c], \tag{20}$$

this constraint can be transformed into the following quadratic penalty function,

$$\mathcal{P}_{en} = p_{en} \sum_{i=1}^{N_c} \left(\sum_{\ell=p_i+N_i^{\min}}^{p_i+N_i^{\max}} x_{i,\ell} - 1 \right)^2, \tag{21}$$

where p_{en} is a constant penalty. Therefore, the total cost function we need to minimize is

$$Cost = f + \mathcal{P}_{con} + \mathcal{P}_{en}. \tag{22}$$

For the total cost function, we must choose an acceptable penalty. If the penalty is too large, the total cost function is only slightly perturbed, but if the penalty is too small, we will probably end up with a solution that does not work. If we let $n = \sum_{i=1}^{N_c} (N_i^{\max} - N_i^{\min} + 1)$, then

$$\sum_{i=1}^{N_c} \sum_{\ell=p_i+N_i^{\min}}^{p_i+N_i^{\max}} x_{i,\ell} = \sum_{k=1}^n x_k, k \in [1, n], \tag{23}$$

that is, $x_{i,\ell}$ is equivalent to x_k , where

$$\begin{aligned} \ell &\in [p_i + N_i^{\min}, p_i + N_i^{\max}], \\ k &\in \left(\sum_{j=1}^{i-1} (N_j^{\max} - N_j^{\min} + 1), \sum_{j=1}^i (N_j^{\max} - N_j^{\min} + 1) \right], \end{aligned} \tag{24}$$

for aircraft $i \neq 1$. When $i = 1$, We have $\ell \in [p_1 + N_1^{\min}, p_1 + N_1^{\max}]$ and $k \in [1, N_1^{\max} - N_1^{\min} + 1]$. In particular, if the maximum system delay allowed by the aircraft and the early takeoff time are the same, they are N^{\max} and N^{\min} , respectively. Correspondingly, for $k \in [1, n]$, we can express the serial number i of the aircraft and the discretized ℓ with k as follows:

$$\begin{aligned} i(k) &= \lceil \frac{k}{N^{\max} - N^{\min} + 1} \rceil, \\ \ell(k) &= (k - 1) \bmod (N^{\max} - N^{\min} + 1) + p_{i(k)} + N^{\min}. \end{aligned} \tag{25}$$

Therefore,

$$f = \Delta_d \sum_{k=1}^n f(i(k)) \cdot \ell(k) x_k, \tag{26}$$

$$\mathcal{P}_{\text{con}} = \frac{p_{\text{con}}}{2} \sum_{k,k'=1}^n \delta_{i(k),i(k')} \delta_{\ell(k),\ell(k')} x_k x_{k'}, \tag{27}$$

$$\mathcal{P}_{\text{en}} = p_{\text{en}} \left(\sum_{k,k'=1}^n \delta_{k,k'} x_k x_{k'} - \sum_{k=1}^n x_k + N_c \right), \tag{28}$$

where

$$\begin{aligned} \delta_{k,k'} &= \begin{cases} 1, & i(k) = i(k') \text{ and } k \neq k' \\ 0, & \text{otherwise} \end{cases}, \\ \delta_{i(k),i(k')} &= \begin{cases} 1, & i(k)i(k') \in \mathcal{R}_m \\ 0, & \text{otherwise} \end{cases}, \\ \delta_{\ell(k),\ell(k')} &= \begin{cases} 1, & -\frac{k-k'}{|k-k'|} \Delta_d (\ell(k) - \ell(k')) \in \mathcal{B}_m \\ 0, & \text{otherwise} \end{cases}. \end{aligned} \tag{29}$$

In addition, we can also remove the constant offset in formula (28) to transform it into

$$\begin{aligned} \mathcal{P}_{\text{en}} &= p_{\text{en}} \left(\sum_{k,k'=1}^n \delta_{k,k'} x_k x_{k'} - \sum_{k=1}^n x_k + N_c \right) - p_{\text{en}} \cdot N_c \\ &= p_{\text{en}} \left(\sum_{k,k'=1}^n \delta_{k,k'} x_k x_{k'} - \sum_{k=1}^n x_k \right). \end{aligned} \tag{30}$$

The total cost function can also be expressed as

$$Q(X) = f + \mathcal{P}_{\text{con}} + \mathcal{P}_{\text{en}} - p_{\text{en}} \cdot N_c, X = (x_1, x_2, \dots, x_n). \tag{31}$$

3. Ising Model and Variational Quantum Algorithm

3.1. Ising Model and QUBO Problem

As one of the most common research areas in quantum computing and quantum annealing in the NISQ era, combinatorial optimization problems can usually be represented as quadratic unconstrained binary optimization (QUBO) problems that are easily solved by quantum computers. Researchers in combinatorial optimization have investigated the QUBO problem since the 1960s. A QUBO problem with n binary decision variables x can be expressed as

$$\min_{x \in \{0,1\}^n} x^T Q x + b^T x, \tag{32}$$

where $Q \in \mathbb{R}^{n \times n}$ is a matrix with symmetry, and $b \in \mathbb{R}^n$ is a vector. Because the binary variable satisfies $x_i = x_i^2$, b can be added to the diagonal of Q . The symmetry of the matrix Q can always be maintained. The QUBO model has become one of the crucial elements in the NISQ era of combinatorial optimization problem solutions due to its characteristics and suitability for the Ising model [15,21]. In the beginning, the Ising model was a mathematical representation of ferromagnetism within statistical mechanics. The model is made up of several discrete variables that were initially described as the direction of the microscopic magnetic momentum associated with the atom's "spin", which can be in two states of +1 or -1. Therefore, the QUBO problem and the Ising model are highly correlated, and an Ising model

$$\min_{z \in \{-1,+1\}^n} z^T J z + h^T z \tag{33}$$

can be obtained by transforming the QUBO problem (32) through a simple mapping as follows:

$$x_i = \frac{1 + z_i}{2}, z_i \in \{-1, +1\}. \tag{34}$$

For a quantum system containing n qubits, we can convert the Ising model (33) into the Hamiltonian after using the Pauli Z-matrix σ_Z^i to replace z_i [22]. Specifically, we can consider z_i as acting σ_Z^i on the i th qubit, and similarly $z_i \cdot z_j$ can be expressed using the tensor product $\sigma_Z^i \otimes \sigma_Z^j$. The eigenvalue of σ_Z^i is $+1$ and -1 , which corresponds to the spin direction of the Ising model (33).

3.2. Variational Quantum Eigensolver (VQE)

The solution of the ground state E of a certain Hamiltonian can usually be solved by VQE, which is the first variational quantum algorithm that was proposed [14] and originally applied to the estimation of the energy of molecular ground states in quantum chemistry. In VQE, according to the Rayleigh–Ritz principle, given a Hamiltonian H whose minimum eigenvalue λ_{\min} and ground state $|\psi_{\min}\rangle$ are unknown, then

$$\lambda_{\min} \leq \lambda_{\theta} = \langle \psi(\theta) | H | \psi(\theta) \rangle, \tag{35}$$

where $|\psi(\theta)\rangle = U(\theta)|\psi_0\rangle$ is the result of the initial state $|\psi_0\rangle$ after the parameterized quantum circuit $U(\theta)$, which represents the eigenstate corresponding to λ_{θ} , and $|\psi(\theta)\rangle$ is an estimate of $|\psi_{\min}\rangle$. The cost function $C(\theta)$ is defined as:

$$C(\theta) = \langle \psi(\theta) | H | \psi(\theta) \rangle. \tag{36}$$

As shown in Figure 2a, we use the cost function $C(\theta)$ to optimize the parameter θ through a classical optimizer and continue to iterate until convergence is reached. Specifically, the parameterized quantum circuits (PQC) we use in this paper are the commonly used hardware efficient ansatz, which is able to somewhat reduce the circuit depth required to implement $U(\theta)$ [23,24]. For a quantum circuit consisting of n qubits, we first apply the RY-gate once to each qubit, then apply the CNOT-gate (i controls j) to all qubits satisfying $i < j$, and finally apply the R-gate again to each qubit, repeating the last two steps p times, where p is the number of layers. We also plotted a quantum circuit of a p -layer PQC containing four qubits (Figure 2b).

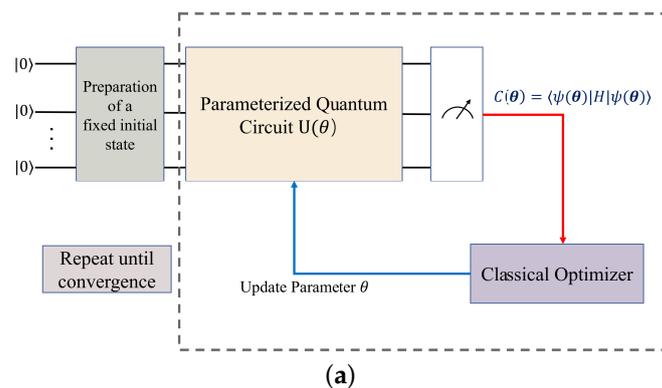


Figure 2. Cont.

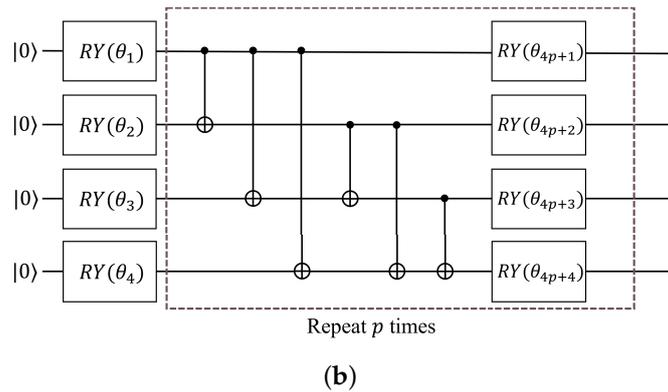


Figure 2. (a) A VQE circuit diagram. (b) An example of a hardware efficient ansatz for a p -layer with four qubits.

3.3. Quantum Approximate Optimization Algorithms

QAOA is another variational quantum algorithm proposed after VQE [5], which was influenced by quantum adiabatic algorithms and initially utilized to approximately solve combinatorial problems [25–27]. The combinatorial optimization problem is defined as n binary strings $x = (x_1, \dots, x_n)$, and the goal of the problem is to maximize or minimize the classical objective function $f(x)$. By transforming the classical binary variable x_i into the Pauli spin $-1/2$ operator σ_i^z , QAOA maps $f(x)$ to the problem Hamiltonian H_p . QAOA follows the ground state evolution process as

$$H(t) = (1 - t)H_M + tH_p, t \in [0, 1]. \tag{37}$$

Usually, the problem Hamiltonian H_p can also be expressed in the form of an Ising Hamiltonian

$$H_p = \sum_{i=1}^n h_i \sigma_Z^i + \sum_{i,j=1}^n J_{ij} \sigma_Z^i \otimes \sigma_Z^j, \tag{38}$$

and the mixer Hamiltonian is defined as

$$H_M = \sum_{i=1}^n \sigma_X^i, \tag{39}$$

where h_i, J_{ij} are real coefficients, and σ_i^z and σ_i^x are the Pauli-Z/X operators applied to the i -th qubit, respectively.

What is used in QAOA is an alternating structure ansatz, often referred to as the quantum alternation operator ansatz [28]; the adiabatic evolution is replaced by p alternate time propagations between H_p and H_M . As shown in Figure 3, in the p -layer QAOA, the initial state is $|+\rangle^{\otimes n}$, the evolution time interval is treated as a variational parameter, i.e., $\theta = \{\gamma, \beta\}$, and the two Hamiltonians are applied alternately p times. The variational form of QAOA is defined as

$$U(\beta, \gamma) = \left[\prod_{i=1}^p U_P(\beta_i) U_M(\gamma_i) \right] H^{\otimes n}, i \in 1, 2, \dots, p, \tag{40}$$

where

$$\begin{aligned} U_P(\gamma_i) &= e^{-i\gamma_i H_p} = e^{-i\gamma_i \left(\sum_{i=1}^n h_i \sigma_Z^i + \sum_{i,j=1}^n J_{ij} \sigma_Z^i \otimes \sigma_Z^j \right)}, \\ U_M(\beta_i) &= e^{-i\beta_i H_M} = e^{-i\beta_i \sum_{i=1}^n \sigma_X^i}, \end{aligned} \tag{41}$$

The quantum state generated after ansatz is

$$|\gamma, \beta\rangle = U(\gamma, \beta)|0\rangle. \tag{42}$$

Similarly, the cost function of QAOA is defined as:

$$C(\gamma, \beta) = \langle \gamma, \beta | H_P | \gamma, \beta \rangle. \tag{43}$$

The evaluation of the cost function is done by repeatedly measuring $|\gamma, \beta\rangle$ on a quantum computer and then optimizing and adjusting $\{\gamma, \beta\}$ with the help of a classical optimizer until an acceptable $\{\gamma, \beta\}$ is found.

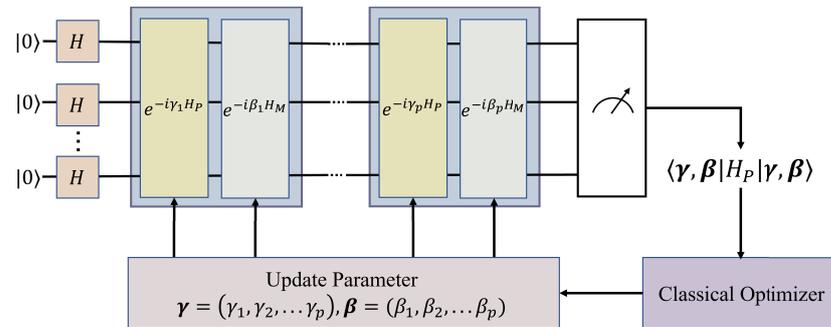


Figure 3. QAOA circuit diagram.

3.4. Conditional Value-at-Risk

For either VQE or QAOA, we cannot directly obtain the expected value of the cost function (Equations (36) and (43)). Taking VQE as an example, we cannot get all information of $|\psi(\theta)\rangle$ by just one measurement. However, we can obtain an estimate of the cost function by multiple measurements. Specifically, we can perform one measurement for n -qubits and get a measurement sample $q_0 q_1 \dots q_{n-1}$. For this sample of measurements, we can write quantum states $|\psi_k(\theta)\rangle = |q_0 q_1 \dots q_{n-1}\rangle$ and then calculate $E_k(\theta) = \langle \psi_k(\theta) | H | \psi_k(\theta) \rangle$. Then, the whole sample after K times measurement can be expressed as $E_k(\theta) (k = 1, \dots, K)$, and the expected value of Equation (36) is the mean value of the sample

$$\frac{1}{K} \sum_{k=1}^K E_k(\theta). \tag{44}$$

To improve the efficiency of VQE and QAOA, we attempt to use CVaR as their objective function [4]. Given a cumulative density function $F(X)$ of a random variable X , the CVaR of X is defined as the expected value of X when $X < F^{-1}(\alpha)$ when the confidence interval is α , that is

$$CVaR_\alpha(X) = \mathbb{E} [X | X \leq F_X^{-1}(\alpha)], \alpha \in [0, 1]. \tag{45}$$

Shortly, CVaR is the expected value on the left-hand side of the distribution of X . Suppose that results of K times measurement are arranged in increasing order from smallest to largest as $\{E_1, \dots, E_K\}$, then the corresponding $CVaR_\alpha$ is

$$CVaR_\alpha = \frac{1}{\lceil \alpha K \rceil} \sum_{k=1}^{\lceil \alpha K \rceil} E_k. \tag{46}$$

The value of CVaR is the mean of the sample, which is the standard objective function when $\alpha = 1$.

4. Result

In this section, we study four examples: (2,1) with $N^{max} = 2, N^{min} = 0$ (Figure 4a), (3,3) (Figure 4b), (4,5) (Figure 4c), and (2,1) with $N^{max} = 2, N^{min} = -2$ (Figure 4d), where (n, k) denotes k pairs of conflict among n aircraft. We use the qasm simulator of IBM’s qiskit to solve the collision avoidance problem by VQE and QAOA. We set $\Delta_d = 1, p_{con} = p_{en} = 12$. In the first three examples, the allowable scheduling delay of each

flight is set as $N^{max} = 2, N^{min} = 0$, so the number of qubits required is $n \cdot (N^{max} + 1)$. For the last example, we set $N^{max} = 2$ and $N^{min} = -2$, and the number of qubits required is $n \cdot (N^{max} - N^{min} + 1)$.

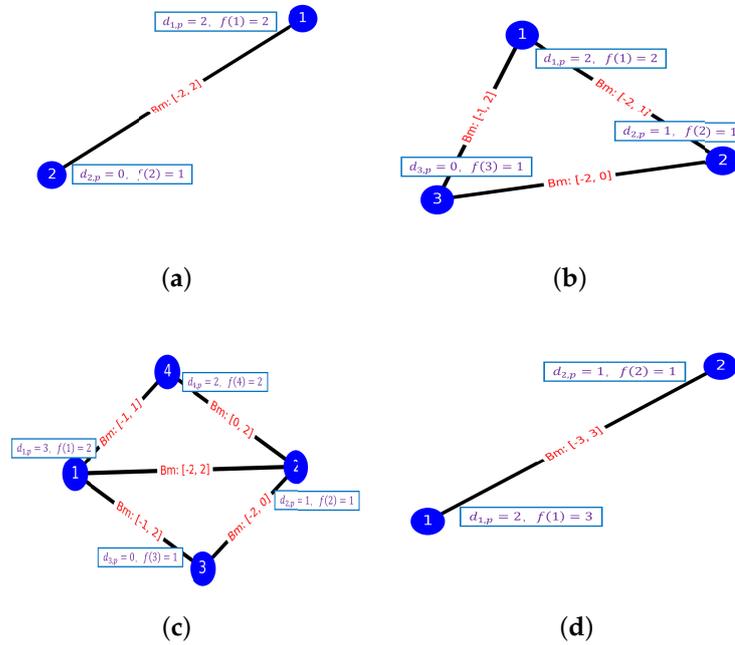


Figure 4. (a) The aircraft conflict information of (2,1) with $N^{max} = 2, N^{min} = 0$. (b) The aircraft conflict information of (3,3). (c) The aircraft conflict information of (4,5). (d) The aircraft conflict information of (2,1) with $N^{max} = 2, N^{min} = -2$. The $d_{i,p}$ and $f(i)$, respectively, represents the pre-takeoff preparation time and weight of flight i ($i = 1, 2, 3, 4$).

In the solution process, first we encode the problem in the form of Equation (31), then map to get the Ising Hamiltonian H_p , and then use VQE or QAOA to find the ground state. We use the gradient-free optimizer linear approximation constrained optimization (COBYLA) optimizer to optimize the variable parameters of PQC for the solution of all four examples. For each example, we tested the performance of QAOA versus VQE for $p = 1, 2$ and $\alpha \in \{0.05, 0.1, 0.25, 0.5, 1\}$.

To compare the two variational quantum algorithms for different values of depth p and α , we also plot the number of iterations of the classical optimizer (COBYLA) against the value of the objective function under each example. In addition, we also plot the ground state sampling probabilities of the Hamiltonian during the iterative process:

$$P_{|\psi_g\rangle} = |\langle \psi | \psi_g \rangle|^2. \tag{47}$$

where $|\psi\rangle$ is the quantum state after the quantum circuit and $|\psi_g\rangle$ is the ground state of the Hamiltonian. We intend that the ground state probability $P_{|\psi_g\rangle}$ obtained by the algorithm should be as large as possible, or at least larger than our given α value.

4.1. Example 1

In the first example, we successfully solved example (2,1) shown in Figure 4a, whose route diagram is shown in Figure 5a. As this problem is relatively small in size, we use only six qubits to encode the problem. The lowest energy is -18 , i.e., the minimum value of Equation (31), corresponding to $X = (0, 1, 0, 1, 0, 0)$, which is the optimal solution to the problem, and the total delay for each aircraft is $D_1 = 3, D_2 = 0$. In Figure 5b, we show the optimized route corresponding to the optimal solution.

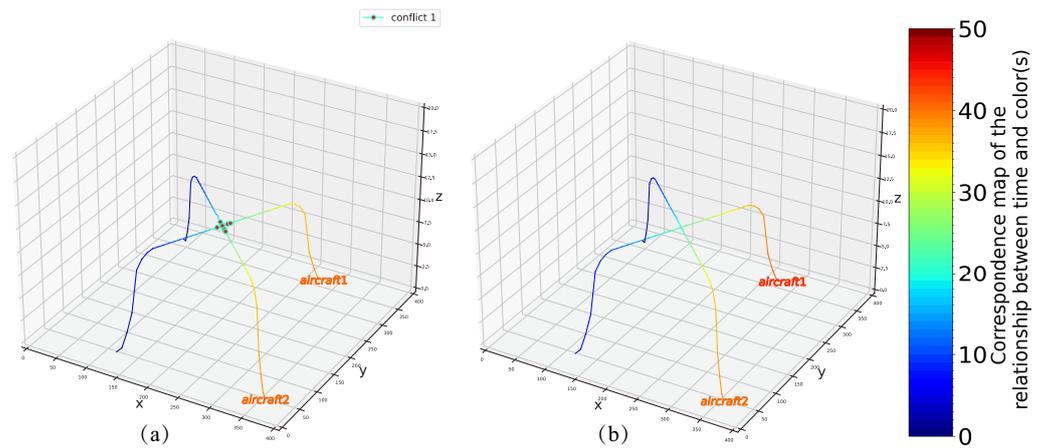


Figure 5. (a) Initial planning route for example 1. (b) The optimized conflict-free route in example 1.

4.2. Example 2

In the second test, we try to solve example (3,3) (as shown in Figure 4b), corresponding to the route in Figure 6a, where we use nine qubits to encode the problem. Finally, we get the optimized route corresponding to the optimal solution (Figure 6b), when $X = (0, 1, 0, 1, 0, 0, 1, 0, 0)$, the minimum energy is -29 and the total delay for each aircraft is $D_1 = 3, D_2 = 1, D_3 = 0$.

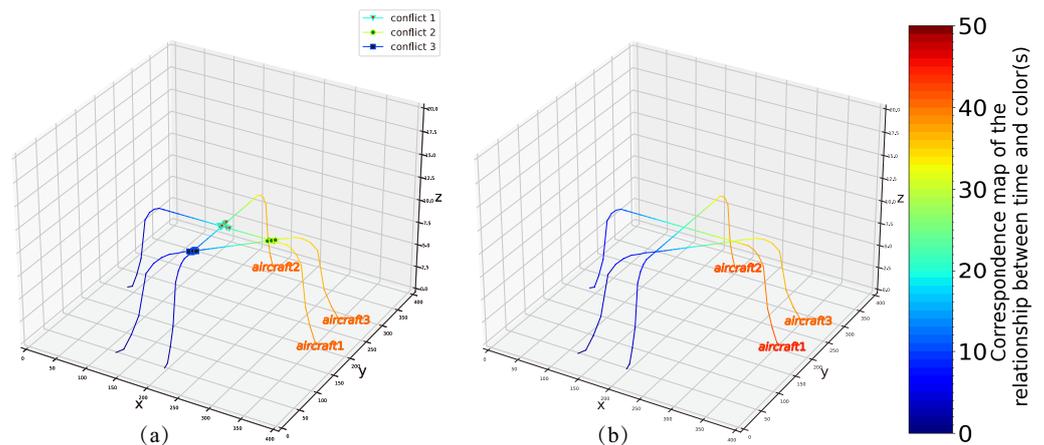


Figure 6. (a) Initial planning route for example 2. (b) The optimized conflict-free route in example 2.

4.3. Example 3

Here, we use 12 qubits to solve the collision example of four aircraft containing five pairs of conflict, corresponding to the details of the conflicts and the routes shown in Figures 4c and 7a. We obtain a minimum energy of -35 , the optimal solution is $X = (0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0)$, and the total delay time of the four aircraft are $D_1 = 3, D_2 = 1, D_3 = 0, D_4 = 2$ (the optimized routes are shown in Figure 7b).

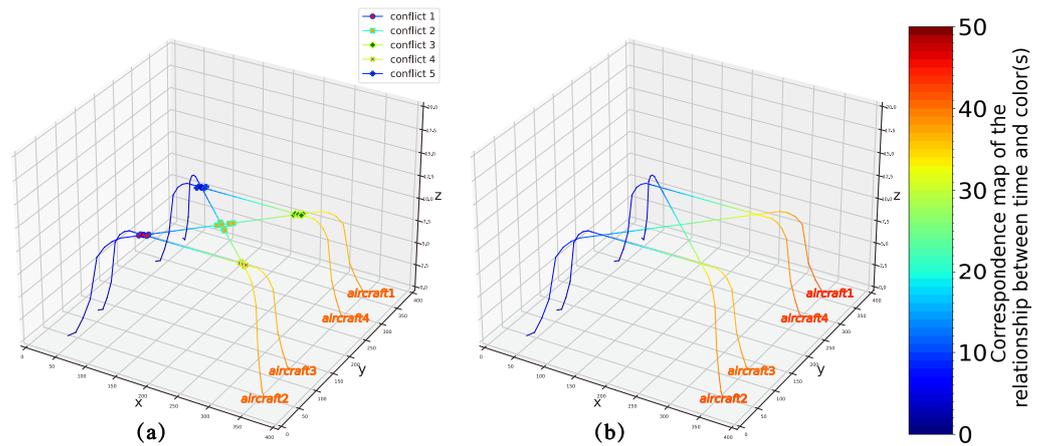


Figure 7. (a) Initial planning route for example 3. (b) The optimized conflict-free route in example 3.

4.4. Example 4

In the last example, we solve another example of the (2, 1) problem using 10 qubits (details of the conflict are shown in Figure 4d). Unlike example 1, in this example we take $N^{min} = -2$ to test the availability of the model, whereas N^{max} is still set as 2. The minimum energy obtained by two algorithms separately is -16 and $X = [0, 0, 0, 1, 0, 1, 0, 0, 0, 0]$, while the total delay of the two aircraft is $D_1 = 3$ and $D_2 = -1$, respectively. Similarly, we have plotted the routes of the two aircraft before and after optimization (Figure 8).

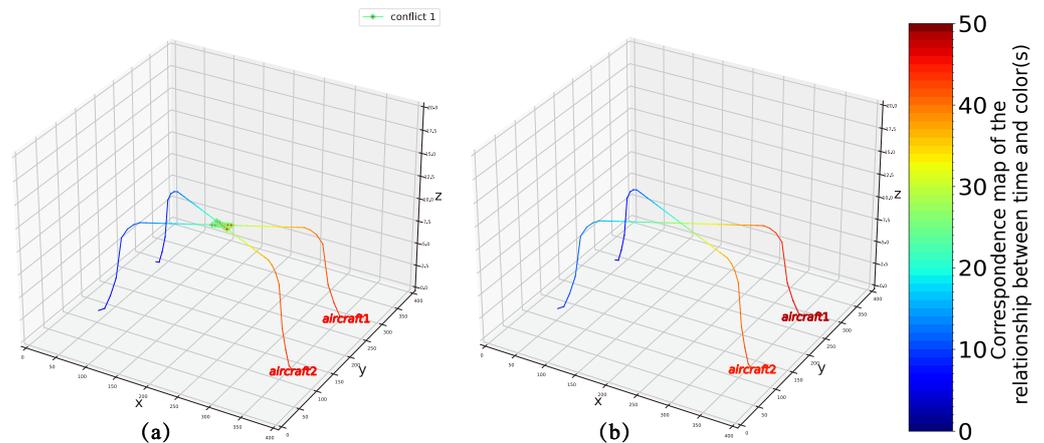


Figure 8. (a) Initial planning route for example 4. (b) The optimized conflict-free route in example 4.

In the following discussion, we write VQE and QAOA that use CVaR as the objective function ($\alpha < 1$) as CVaR-VQE and CVaR-QAOA, respectively.

We first compare the objective functions of the original VQE and CVaR-VQE during the iterative process. In Figures 9 and 10, we plot the probability of the objective function values against the optimal solution, respectively. It is obvious from Figure 9 that when we choose $\alpha < 1$, the number of iterations required to converge the objective function value to the ground state energy is significantly lower than the original VQE. To take example 1, which uses six qubits, in Figure 9a, the original VQE does not reach the ground state energy until the 100th iteration, whereas even the slowest convergence, $\alpha = 0.5$, reaches convergence at about the 70th iteration. And as the number of qubits increases, the performance of CVaR-VQE is still superior to the original VQE, although it also decreases. For the ground state probability $P_{|\psi_g\rangle}$, the number of iterations required for $P_{|\psi_g\rangle}$ to converge increases as the α value rises, but even in example 3, which has the highest number of iterations, CVaR-VQE still has an advantage in the number of iterations required for the ground state probabilities and the objective function value to reach convergence.

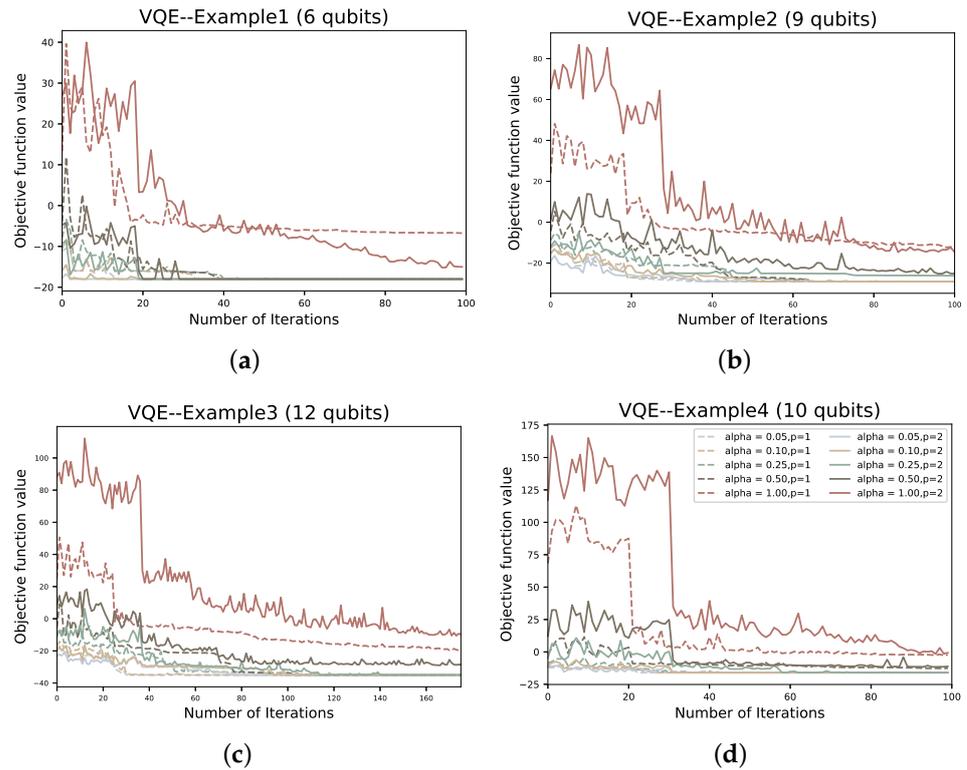


Figure 9. The objective function values corresponding to the number of iterations of VQE at different α and p in the four examples; (a–d) represent examples 1–4, respectively.

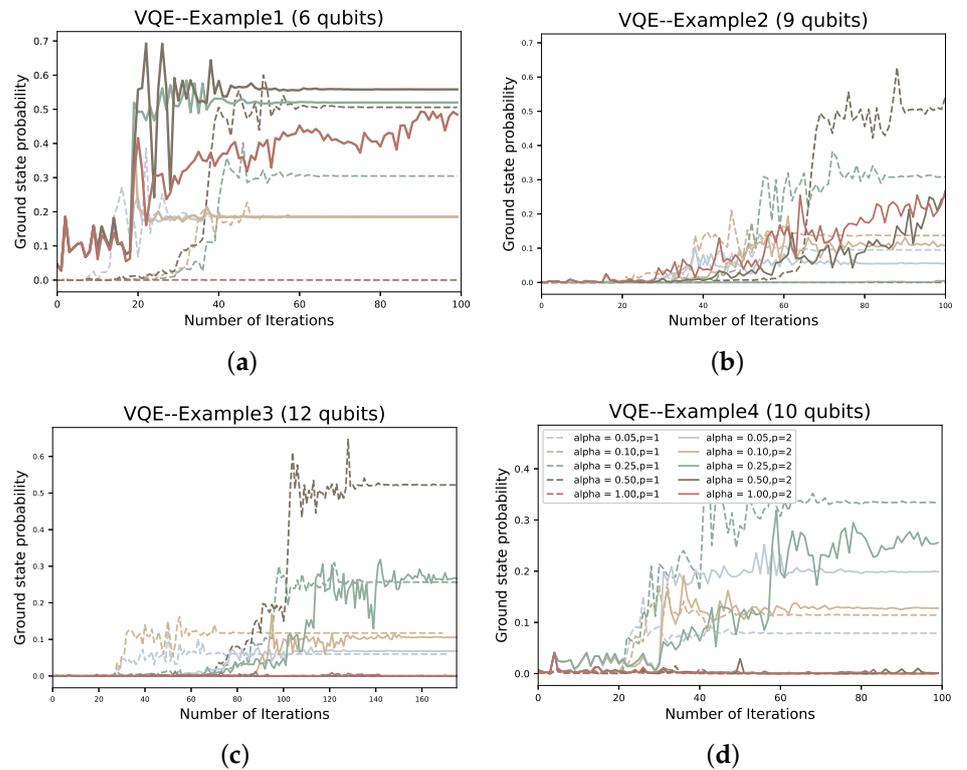


Figure 10. The ground state probability corresponding to the number of iterations of VQE at different α and p in the four examples; (a–d) represent examples 1–4, respectively.

As the depth p increases and α decreases, we find that there is a certain improvement in both the objective function value and the ground state probability $P_{|\psi_g\rangle}$ of QAOA

(as shown in Figures 11 and 12). However, overall, the convergence rate of QAOA is still very slow and the ground state probability is generally very low, with the highest $P_{|\psi_g\rangle}$ still less than 0.15. When we optimize, most QAOA ends when the number of iterations reaches 50. Compared to the VQE, the performance of QAOA is inferior. Without considering the impact of the classical optimizer, one of the important reasons for our analysis is that the hardware efficient ansatz is used in the VQE. A VQE of depth p contains $n(p + 1)$ variational parameters and additional control gates, whereas the QAOA has only $2p$ variational parameters. Therefore, if we want to improve the performance of QAOA, increasing the depth p is an effective way. However, this will also inevitably lead to an increase in the solution time. Due to the current NISQ devices, it is still difficult to successfully implement quantum circuits with excessive depth. It has been shown in some previous studies that when NISQ devices run quantum algorithms, the fidelity of the quantum state prepared by the quantum program and the minimum expected value of the Hamiltonian will be affected because of the existence of noise, which makes the results have deviations [29].

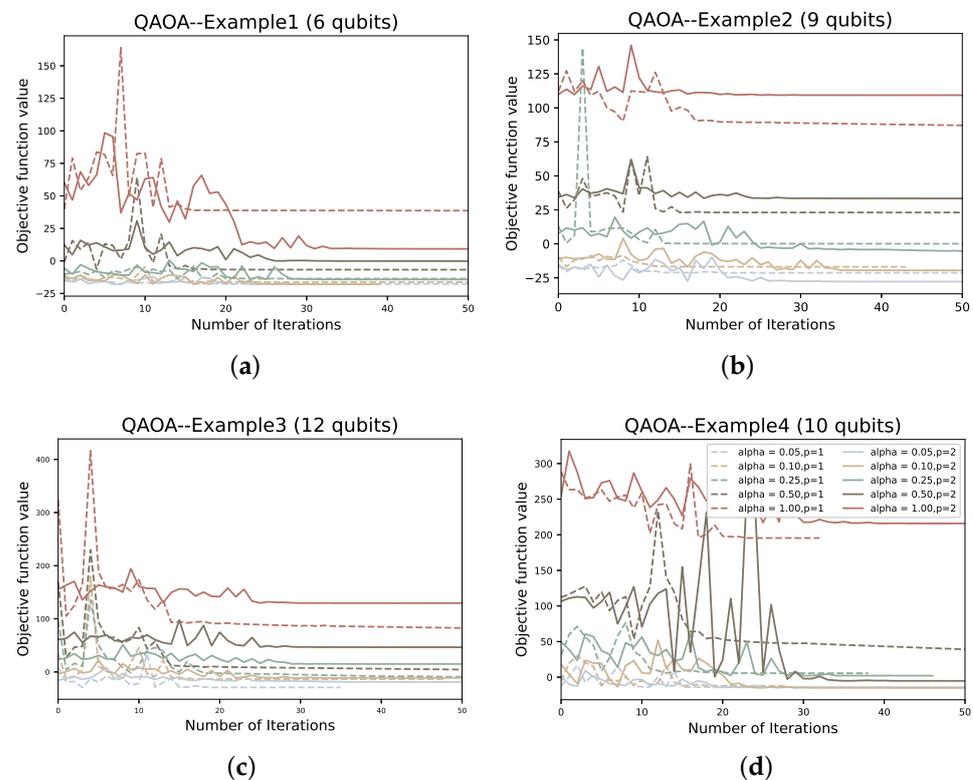


Figure 11. The objective function values corresponding to the number of iterations of QAOA at different α and p in the four examples; (a–d) represent examples 1–4, respectively.

Finally, in order to compare the solution results, we plot the probability of obtaining the optimal solution (Figure 13) and the feasible solution (Figure 14) using VQE and QAOA in solving the four examples. From the resulting probability of optimal solution, the results of the VQE solution also confirm to some extent that the performance of VQE decreases as the α value increases. Taking the most remarkable example 3 as an example, the probability of obtaining the optimal solution even differs by 0.4 between the solution results of the original VQE and the CVaR-VQE (at $\alpha = 0.05$) when $p = 2$. From the results of QAOA, the accuracy of its solution is indeed lower than that of VQE to a certain extent. In terms of the probability of feasible solutions obtained from the QAOA solution, both the application of CVaR and the increase of depth p have improved the performance of QAOA. Recalling our previous discussion of the existence of QAOA with small ground state probability $P_{|\psi_g\rangle}$ at iteration, we believe that this could be determined by the characteristics of QAOA

itself—to obtain an approximate solution to the combinatorial problem rather than an optimal solution, i.e., the quantum state be relatively flat, and this phenomenon was studied in recent work [4]. This means that the impact of the CVaR strategy we employ for QAOA is likely to be positive but relatively small. The probability of feasible solutions obtained by solving VQE and QAOA (as shown in Figure 14) also demonstrates the positive impact of CVaR on VQE and QAOA from the side.

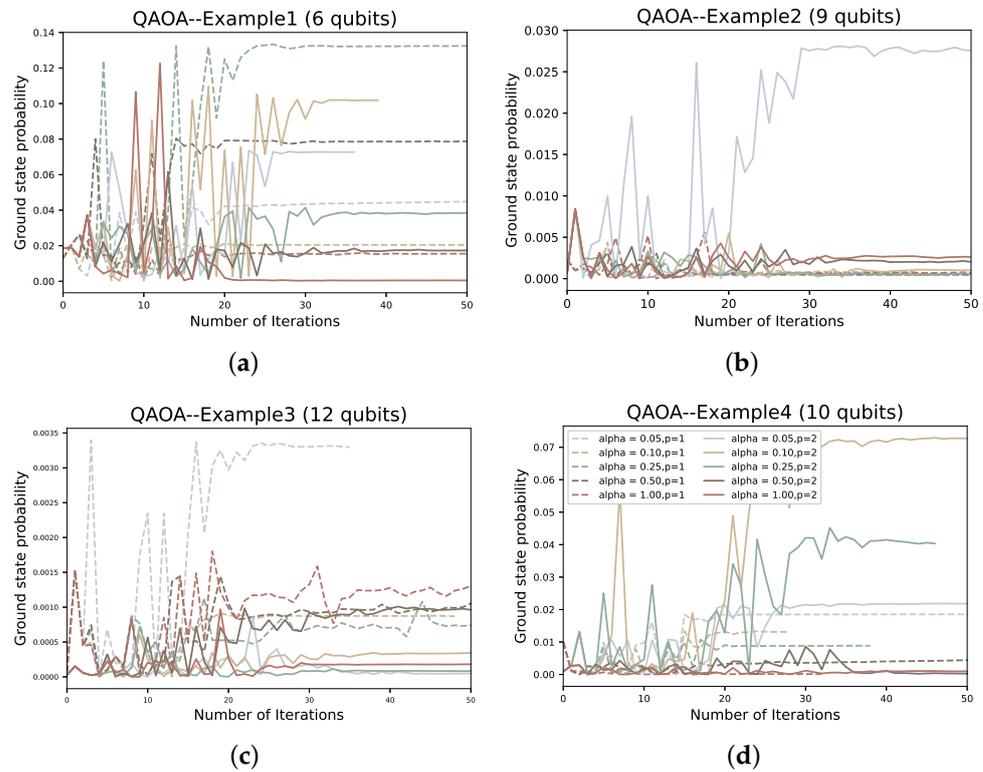


Figure 12. The ground state probability corresponding to the number of iterations of QAOA at different α and p in the four examples; (a–d) represent examples 1–4, respectively.

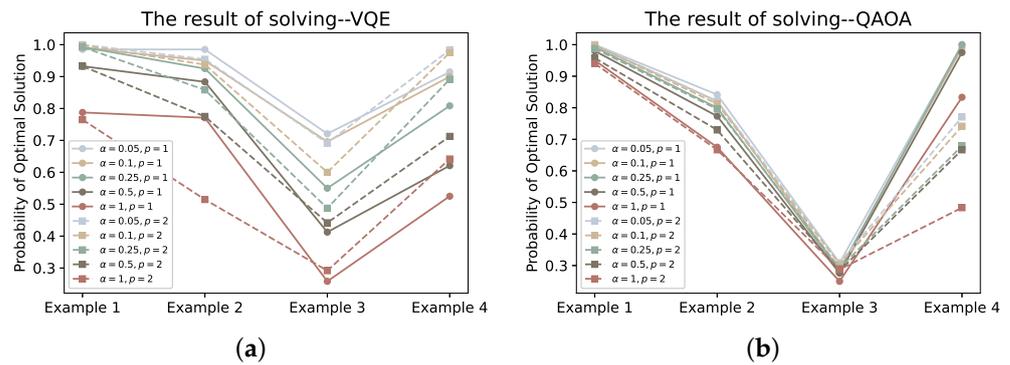


Figure 13. Subfigures (a,b) plot the optimal solution probabilities obtained using VQE and QAOA for different depths p and α values, respectively.

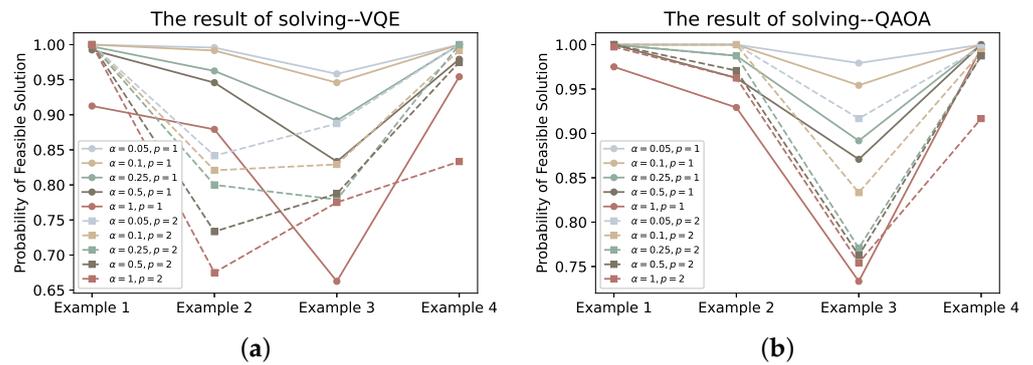


Figure 14. Subfigures (a,b) plot the feasible solution probabilities obtained using VQE and QAOA for different depths p and α values, respectively.

5. Conclusions

In this paper, we propose a model for the aircraft collision avoidance problem and give a mapping of the model to the QUBO problem. First, the potential conflicts are formulated as mutually disjoint conflict sets. Subsequently, in order to encode the problem as a QUBO problem, we discretize the configuration space of the problem. After encoding the problem as a Hamiltonian, we use two variational quantum algorithms (VQE and QAOA) to solve the ground state of the Hamiltonian of the problem, respectively. In the experimental section, we test these two variational quantum algorithms in four different examples and compare the performance of the two algorithms under different α values and variational form depth p .

CVaR had a positive impact on VQA, because both CVaR-VQE and CVaR-QAOA have improved the performance of the algorithm. For the four examples we have solved, the probability of the resulting feasible solution is still approximately 70%, even for the worst solution result, and the best solution is even more than 95%. Of course, the probability of the best solution is relatively low, and the worst solution result is only approximately 30%. In conclusion, if we adjust the value of α appropriately, the probability of obtaining a feasible solution to the problem can be raised to a fairly high level. In real applications, we can obtain feasible and optimal solutions by repeatedly running the QAOA or VQE algorithms, but this also means that we must choose between the quality of the solution and the running time when solving.

The focus of this paper is on the collision avoidance problem of aircraft and the variational quantum algorithm used for this problem. For future work, we suggest that further research can be conducted in two ways.

- In model building, the main consideration of this paper is to avoid collision by extending the aircraft takeoff time. In a real situation, in order to extend the usability of the model, we can also consider making some adjustments to the flight path to avoid collision, including changing the flight speed in a specific interval and partially changing the flight path of the aircraft, etc. These strategies can be collectively referred to as maneuvering collision avoidance strategies. We can adjust the cost function of the model by mapping the maneuver cost to the time delay cost in conjunction with the actual demand.
- In the aspect of quantum computing, on the one hand, in order to improve the efficiency of model solving, we can use more efficient coding methods to reduce the number of qubits required and the depth of quantum circuits to compress the time to solve the problem when encoding the decision variables. However, this requires us to modify the mapping between the model and the QUBO problem. Second, from the algorithmic point of view, improving the classical optimizer or adopting a more efficient ansatz structure is also an effective way to improve the efficiency of the algorithm.

Author Contributions: Conceptualization, Z.H.; methodology, Q.L.; software, Z.H.; validation, Z.H. and Q.L.; formal analysis, J.Z.; investigation, M.S.; resources, Z.H.; data curation, J.Z.; writing—original draft preparation, Z.H. and Q.L.; writing—review and editing, Q.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

VQA	Variational Quantum Algorithms
VQE	Variational Quantum Eigensolver
QAOA	Quantum Approximate Optimization Algorithm
QUBO	Quadratic Unconstrained Binary Optimization
ATM	Air Traffic Management
ATFM	Air Traffic Flow Management
UAV	Unmanned Aerial Vehicles
UTM	Unmanned Traffic Management
NISQ	Noisy Intermediate-Scale Quantum
CVaR	Conditional Value-at-Risk
UAM	Urban Air Mobility
NASA	National Aeronautics and Space Administration
PQC	Parameterized Quantum Circuits

References

- Hildmann, H.; Kovacs, E. Using unmanned aerial vehicles (UAVs) as mobile sensing platforms (MSPs) for disaster response, civil security and public safety. *Drones* **2019**, *3*, 59. [\[CrossRef\]](#)
- Ho, F.; Geraldles, R.; Goncalves, A.; Cavazza, M.; Prendinger, H. Improved conflict detection and resolution for service UAVs in shared airspace. *IEEE Trans. Veh. Technol.* **2018**, *68*, 1231–1242. [\[CrossRef\]](#)
- Stollenwerk, T.; O’Gorman, B.; Venturelli, D.; Mandra, S.; Rodionova, O.; Ng, H.; Sridhar, B.; Rieffel, E.G.; Biswas, R. Quantum annealing applied to de-conflicting optimal trajectories for air traffic management. *IEEE Trans. Intell. Transp. Syst.* **2019**, *21*, 285–297. [\[CrossRef\]](#)
- Barkoutsos, P.K.; Nannicini, G.; Robert, A.; Tavernelli, I.; Woerner, S. Improving variational quantum optimization using CVaR. *Quantum* **2020**, *4*, 256. [\[CrossRef\]](#)
- Farhi, E.; Goldstone, J.; Gutmann, S. A quantum approximate optimization algorithm. *arXiv* **2014**, arXiv:1411.4028.
- Arute, F.; Arya, K.; Babbush, R.; Bacon, D.; Bardin, J.C.; Barends, R.; Biswas, R.; Boixo, S.; Brandao, F.G.; Buell, D.A.; et al. Quantum supremacy using a programmable superconducting processor. *Nature* **2019**, *574*, 505–510. [\[CrossRef\]](#)
- Zhong, H.S.; Wang, H.; Deng, Y.H.; Chen, M.C.; Peng, L.C.; Luo, Y.H.; Qin, J.; Wu, D.; Ding, X.; Hu, Y.; et al. Quantum computational advantage using photons. *Science* **2020**, *370*, 1460–1463. [\[CrossRef\]](#)
- Zhong, H.S.; Deng, Y.H.; Qin, J.; Wang, H.; Chen, M.C.; Peng, L.C.; Luo, Y.H.; Wu, D.; Gong, S.Q.; Su, H.; et al. Phase-programmable gaussian boson sampling using stimulated squeezed light. *Phys. Rev. Lett.* **2021**, *127*, 180502. [\[CrossRef\]](#)
- Preskill, J. Quantum computing in the NISQ era and beyond. *Quantum* **2018**, *2*, 79. [\[CrossRef\]](#)
- Cerezo, M.; Arrasmith, A.; Babbush, R.; Benjamin, S.C.; Endo, S.; Fujii, K.; McClean, J.R.; Mitarai, K.; Yuan, X.; Cincio, L.; et al. Variational quantum algorithms. *Nat. Rev. Phys.* **2021**, *3*, 625–644. [\[CrossRef\]](#)
- Farhi, E.; Harrow, A.W. Quantum supremacy through the quantum approximate optimization algorithm. *arXiv* **2016**, arXiv:1602.07674.
- Zhou, L.; Wang, S.T.; Choi, S.; Pichler, H.; Lukin, M.D. Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices. *Phys. Rev. X* **2020**, *10*, 021067. [\[CrossRef\]](#)
- Harrigan, M.P.; Sung, K.J.; Neeley, M.; Satzinger, K.J.; Arute, F.; Arya, K.; Atalaya, J.; Bardin, J.C.; Barends, R.; Boixo, S.; et al. Quantum approximate optimization of non-planar graph problems on a planar superconducting processor. *Nat. Phys.* **2021**, *17*, 332–336. [\[CrossRef\]](#)
- Peruzzo, A.; McClean, J.; Shadbolt, P.; Yung, M.H.; Zhou, X.Q.; Love, P.J.; Aspuru-Guzik, A.; O’Brien, J.L. A variational eigenvalue solver on a photonic quantum processor. *Nat. Commun.* **2014**, *5*, 1–7. [\[CrossRef\]](#) [\[PubMed\]](#)

15. Glover, F.; Kochenberger, G.; Du, Y. A tutorial on formulating and using QUBO models. *arXiv* **2018**, arXiv:1811.11538.
16. Domino, K.; Koniarczyk, M.; Krawiec, K.; Jałowicki, K.; Deffner, S.; Gardas, B. Quantum annealing in the NISQ era: Railway conflict management. *arXiv* **2021**, arXiv:2112.03674.
17. Kellermann, R.; Biehle, T.; Fischer, L. Drones for parcel and passenger transportation: A literature review. *Transp. Res. Interdiscip. Perspect.* **2020**, *4*, 100088. [[CrossRef](#)]
18. Ayamga, M.; Akaba, S.; Nyaaba, A.A. Multifaceted applicability of drones: A review. *Technol. Forecast. Soc. Chang.* **2021**, *167*, 120677. [[CrossRef](#)]
19. Sumitomo Corporation. Available online: <https://www.sumitomocorp.com/en/jp/news/release/2021/group/14850> (accessed on 2 June 2021).
20. Gipson, L. Available online: <https://www.nasa.gov/aero/nasa-embraces-urban-air-mobility> (accessed on 10 February 2020).
21. Lewis, M.; Glover, F. Quadratic unconstrained binary optimization problem preprocessing: Theory and empirical analysis. *Networks* **2017**, *70*, 79–97. [[CrossRef](#)]
22. Lucas, A. Ising formulations of many NP problems. *Front. Phys.* **2014**, *5*. [[CrossRef](#)]
23. Kandala, A.; Mezzacapo, A.; Temme, K.; Takita, M.; Brink, M.; Chow, J.M.; Gambetta, J.M. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature* **2017**, *549*, 242–246. [[CrossRef](#)] [[PubMed](#)]
24. Gard, B.T.; Zhu, L.; Barron, G.S.; Mayhall, N.J.; Economou, S.E.; Barnes, E. Efficient symmetry-preserving state preparation circuits for the variational quantum eigensolver algorithm. *NPJ Quantum Inf.* **2020**, *6*, 1–9. [[CrossRef](#)]
25. Shaydulin, R.; Safro, I.; Larson, J. Multistart methods for quantum approximate optimization. In Proceedings of the 2019 IEEE high performance extreme computing conference (HPEC), Waltham, MA, USA, 24–26 September 2019; pp. 1–8.
26. Wang, Z.; Hadfield, S.; Jiang, Z.; Rieffel, E.G. Quantum approximate optimization algorithm for MaxCut: A fermionic view. *Phys. Rev. A* **2018**, *97*, 022304. [[CrossRef](#)]
27. Crooks, G.E. Performance of the quantum approximate optimization algorithm on the maximum cut problem. *arXiv* **2018**, arXiv:1811.08419.
28. Hadfield, S.; Wang, Z.; O’gorman, B.; Rieffel, E.G.; Venturelli, D.; Biswas, R. From the quantum approximate optimization algorithm to a quantum alternating operator ansatz. *Algorithms* **2019**, *12*, 34. [[CrossRef](#)]
29. Guerreschi, G.G.; Matsuura, A.Y. QAOA for Max-Cut requires hundreds of qubits for quantum speed-up. *Sci. Rep.* **2019**, *9*, 1–7. [[CrossRef](#)] [[PubMed](#)]