Applying the Quantum Approximate Optimization Algorithm to the UAV Collision Avoidance Problem

Kalleb C. Santos[1], Maria Eduarda W. M. Vianna[1], Mateus K. de Oliveira[1], Evandro Chagas Ribeiro da Rosa[2], Alison R. Panisson[1], Luiz F. Bittencourt[3], Roberto Rodrigues-Filho[4]

[1]Departamento de Computação – Universidade Federal de Santa Catarina

[2]Departamento de Informática e Estatística – Universidade Federal de Santa Catarina

[3]Instituto de Computação – Universidade Estadual de Campinas

[4]Departamento de Ciência da Computação – Universidade de Brasília

kalleb.c.santos@grad.ufsc.br, roberto.filho@unb.br

Abstract. *The increasing availability of quantum hardware has stimulated interest in applying quantum algorithms to combinatorial optimization problems. Collision avoidance for unmanned aerial vehicles (UAVs) is one such challenge due to the computational complexity of coordinating multiple agents in shared airspace. In this paper, we formulate the problem as a Quadratic Unconstrained Binary Optimization (QUBO) model that captures route selection and collision constraints, enabling the use of the Quantum Approximate Optimization Algorithm (QAOA). To improve efficiency, we introduce a constrained QAOA Ansatz employing an XY-mixer that preserves one-hot route assignment constraints and restricts the search space to valid solutions. We evaluate the approach through noiseless simulations and device-based noise models, analyzing its behavior under realistic conditions and highlighting the opportunities and limitations of near-term quantum hardware for multi-agent routing problems.*

1. Introduction

Unmanned Aerial Vehicles (UAVs) are increasingly deployed in a wide range of civil and industrial applications, including infrastructure inspection, logistics, emergency response, and precision agriculture [Valavanis and Vachtsevanos 2014]. Their growing adoption is largely driven by advances in autonomy, sensing capabilities, and communication technologies, which allow UAVs to operate with minimal human intervention. As these systems become more autonomous and begin to share airspace with other UAVs, particularly in dense urban environments, ensuring safe navigation becomes a critical challenge.

One of the central safety concerns in such environments is the prevention of midair collisions. Collision avoidance requires UAVs to continuously reason about their trajectories while accounting for dynamic obstacles, uncertain environmental conditions, and the actions of other autonomous agents. When multiple drones operate simultaneously, the problem becomes significantly more complex because each drone's decision can influence the feasible trajectories of the others. Thus, collision avoidance often entails solving combinatorial problems within strict time constraints [de Oliveira et al. 2024].

Traditional approaches typically address this challenge using classical optimization, search, or multi-agent coordination techniques [Rezaee et al. 2024, Tang et al. 2022]. While these methods have produced effective solutions for small or moderately sized scenarios, the computational complexity of trajectory planning and conflict resolution grows rapidly with the number of drones and environmental constraints. This growth can limit the scalability of classical approaches, particularly in scenarios involving dense drone traffic and highly dynamic environments.

Quantum computing has recently emerged as a promising paradigm for tackling computationally demanding optimization and combinatorial problems [Vianna et al. 2025]. Hybrid classical-quantum algorithms have been proposed as a way to exploit near-term quantum hardware while relying on classical optimization procedures. In this context, quantum algorithms may serve as specialized accelerators capable of exploring large solution spaces more efficiently. As quantum hardware continues to mature and become more accessible, investigating their potential role in complex decision-making systems, such as UAV traffic management, becomes an important research direction.

Therefore, this paper presents two main contributions toward addressing the drone collision avoidance problem using quantum computing. First, we formulate the collision avoidance problem over a predefined set of candidate routes as a Quadratic Unconstrained Binary Optimization (QUBO) Hamiltonian, which enables the use of the Quantum Approximate Optimization Algorithm (QAOA), a hybrid classical-quantum optimization method. Second, because the QUBO construction relies on one-hot encoding, we demonstrate how to reduce the effective search space by employing an XY-mixer within the QAOA ansatz. This architectural modification embeds one of the Hamiltonian constraints directly into the mixer, significantly restricting the feasible solution space explored during the optimization process.

Finally, to facilitate transparency and reproducibility, we make our implementation publicly available[1] so that other researchers can replicate and extend our experiments. The remainder of this paper is organized as follows. Section 2 surveys related work on drone collision avoidance. Section 3 presents the problem definition and QUBO modeling. Section 4 details the classical-quantum optimization framework, including the tailored QAOA implementation and subspace restriction. Section 5 describes the experimental setup and simulation results. Section 6 analyzes the algorithm's scalability and search space reduction. Finally, Section 7 concludes the paper.

2. Related Work

This section reviews relevant work on quantum algorithms applied to collision avoidance and route optimization for drones. In particular, we focus on approaches that model the problem as a combinatorial optimization task and leverage quantum computing techniques to address its computational complexity.

Recent literature has explored the use of quantum algorithms for multi-agent route optimization. Most existing studies that address spatial path selection rely on quantum annealing methods. In these approaches, routing decisions are typically formulated as Quadratic Unconstrained Binary Optimization (QUBO) problems and solved using quantum annealers. For example, [Clark et al. 2019, Haba et al. 2025] model spatial route selection as QUBO instances and evaluate their solutions using annealing-based hardware.

[1]The source code is available at `https://doi.org/10.5281/zenodo.20272749`.

[Haba et al. 2025] further extend this approach to the context of urban air mobility fleets by pre-computing route conflicts through classical processing and mapping the resulting decision problem to a Maximum Weighted Independent Set executed on a D-Wave Advantage processor.

In contrast, gate-based variational quantum algorithms remain relatively underexplored for this problem domain. One notable example is presented in [Huang et al. 2022], where both the Variational Quantum Eigensolver (VQE) and the Quantum Approximate Optimization Algorithm (QAOA) are applied to UAV collision avoidance. In that work, however, spatial trajectories are assumed to be predefined and fixed. The quantum algorithm is therefore used only to optimize temporal scheduling decisions, such as adjusting takeoff windows to avoid aerial conflicts. Furthermore, the study relies on a standard QAOA formulation without introducing modifications to constraint handling.

Our work differs from prior gate-based approaches by applying QAOA directly to spatial route selection rather than temporal scheduling. Candidate routes are generated through classical preprocessing, and spatio-temporal conflicts between route pairs are identified using a tolerance window. Only conflicting pairs are included in the cost Hamiltonian as pairwise exclusion penalties, yielding a sparse operator. We further introduce a constrained QAOA ansatz [Farhi et al. 2014] that enforces the one-hot route assignment constraint using an XY-mixer and W-state initialization, ensuring that only valid assignments are explored and eliminating the need for a corresponding penalty term. We evaluate both the standard and constrained formulations under noiseless simulation and under the IBM Brisbane noise model.

3. QUBO Model

We investigate a route selection problem with collision avoidance constraints within a d-dimensional grid environment. Each vehicle $v \in V$ is defined by its start and goal terminal nodes $(s_v,\ g_v)$ and a set of candidate routes R_v, generated beforehand by an external routing or path planning procedure. For consistency across the decision space, each vehicle is assigned an equal number of candidate routes, such that $|R_i| = |R_j| = |R|,\ \forall i, j \in V$. A valid route is represented as an ordered sequence of nodes $r = (r_0,\ r_1,\ \ldots,\ r_L)$, where $r_0 = s_v$, $r_L = g_v$, and any two consecutive nodes $(r_{k-1},\ r_k)$ must be adjacent, including diagonal neighbors, respecting the grid limits.

Our proposal encapsulates this problem definition within a QUBO formulation. First, Section 3.1 defines the binary decision variables representing each vehicle's possible assignments. We then detail the components of the objective function, specifically the route cost term in Section 3.2 and the one-hot route selection constraint in Section 3.3. Following this, Section 3.4 outlines our temporal modeling assumptions to establish the conflict modeling framework detailed in Section 3.5. Finally, Section 3.6 consolidates these components into the final cost Hamiltonian.

3.1. Decision Variables

To represent the route selection process as a discrete optimization problem, we introduce a set of binary decision variables. For each vehicle $v \in V$ and each candidate route $r \in R_v$, we define the indicator variable $x_{v,r}$ as follows:

$$x_{v,r} = \begin{cases} 1, & \text{if vehicle } v \text{ selects route } r \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

The total qubit overhead is determined by the product of the vehicle set cardinality $|V|$ and the number of candidate routes per vehicle $|R|$, resulting in $|V| \cdot |R|$ variables.

3.2. Route Cost Term

Candidate routes are precomputed and treated as atomic decision options. Assuming a uniform traversal cost across the grid, the cost of a candidate route $r = (r_1, r_2, \ldots, r_L)$ is equal to its length L, representing the number of visited nodes. We define this individual cost as:

$$c_{v,r} = L. \tag{2}$$

Consequently, the total cost contribution to the QUBO objective function is the sum of the costs of all selected routes:

$$H_{\text{cost}} = \sum_{v \in V} \sum_{r \in R_v} c_{v,r} x_{v,r}. \tag{3}$$

3.3. Route Selection Constraint

To ensure a valid assignment, each vehicle $v \in V$ must be allocated exactly one route from its respective candidate set R_v. This "one-hot" requirement is formally as:

$$\sum_{r \in R_v} x_{v,r} = 1. \tag{4}$$

To embed this within the unconstrained QUBO framework, we convert the condition into a quadratic penalty term, H_{route}, which is added to the total objective function:

$$H_{\text{route}} = A \sum_{v \in V} \left(1 - \sum_{r \in R_v} x_{v,r} \right)^2, \tag{5}$$

where $A > 0$ is a penalty coefficient that scales the constraint violation.

3.4. Temporal Modeling Assumptions

To model vehicle dynamics, we assume synchronized, discrete-time motion. Because a route is defined as an ordered sequence of adjacent grid nodes, the positional index of a node inherently represents a discrete time step. This temporal mapping relies on three operational rules: (1) all vehicles depart simultaneously at $t = 0$; (2) vehicles traverse at a constant unit speed, advancing exactly one grid transition per time step without waiting actions; and (3) upon reaching its final node, a vehicle immediately exits the grid. Consequently, route indices directly function as discrete timestamps, enabling temporal conflict detection through simple index comparison.

3.5. Conflict Modeling

Building upon our temporal assumptions, a conflict occurs if two distinct vehicles visit the same grid node within a short temporal window. To account for collision avoidance, we define $\mathcal{C}$, the set of all conflicting vehicle-route assignments. Two candidate routes $r \in R_v$ and $s \in R_u$ are in conflict if they share a spatial node ($r_i = s_j$) within a fixed temporal tolerance parameter Δ:

$$\mathcal{C} = \{((v,\ r),\ (u,\ s)) \mid v \neq u, r \in R_v, s \in R_u : \exists i, j \text{ s.t. } (r_i = s_j) \wedge (|i-j| < \Delta)\}. \quad (6)$$

To prevent the simultaneous selection of mutually exclusive paths, these conflicts are heavily penalized in the objective function:

$$H_{\text{conflict}} = B \sum_{((v,\ r),\ (u,\ s)) \in \mathcal{C}} x_{v,r} x_{u,s}, \quad (7)$$

where $B > 0$ is a penalty scaling parameter.

3.6. Final Cost Hamiltonian

The complete optimization problem is mapped onto a single objective function, the cost Hamiltonian H. This formulation integrates path cost minimization with the required one-hot assignment and collision avoidance penalties:

$$H = \underbrace{\sum_{v \in V} \sum_{r \in R_v} c_{v,r} x_{v,r}}_{\text{route cost}} + \underbrace{A \sum_{v \in V} \left(1 - \sum_{r \in R_v} x_{v,r}\right)^2}_{\text{route constraint}} + \underbrace{B \sum_{((v,\ r),\ (u,\ s)) \in \mathcal{C}} x_{v,r} x_{u,s}}_{\text{collision penalty}}. \quad (8)$$

4. Proposed Classical-Quantum Optimization

This section introduces our approach to solving the drone collision avoidance problem. Section 4.1 outlines the overall architecture of our hybrid computational pipeline. Section 4.2 details the classical pre-processing steps required to prepare the problem instance. Finally, Section 4.3 presents the quantum execution phase, highlighting our tailored QAOA implementation and the proposed subspace restriction.

4.1. Hybrid Framework

We propose a hybrid computational pipeline designed to effectively address the route selection problem, as presented in Figure 1. The architecture enforces a clear separation of concerns: it leverages classical resources for data-intensive pre-processing while reserving the quantum processor for the high-dimensional combinatorial optimization task. The overall workflow comprises defining the vehicle parameters, generating the candidate route sets R_v, and constructing the conflict set $\mathcal{C}$. Once the problem is mathematically mapped, it is solved using a Variational Quantum Algorithm (VQA), specifically QAOA, to identify the ground state configuration of the total Hamiltonian.

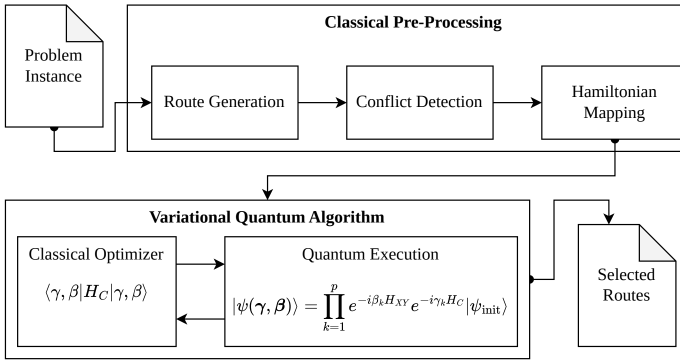


Figure 1. Classical-Quantum Optimization Framework.

4.2. Classical Pre-Processing

The pre-processing stage is executed entirely on classical hardware to efficiently handle the initial computations over the grid-structured graph. This phase is divided into four primary steps:

1. Vehicle Definition: A set of vehicles V is established by defining the start and goal terminal pairs $(s_v,\ g_v)$ for each vehicle within the grid environment.
2. Route Generation: For each vehicle $v \in V$, a set of $|R|$ candidate routes, denoted as R_v, is generated to traverse the spatial grid from s_v to g_v.
3. Conflict Detection: A systematic assessment of the grid coordinates across all route pairs is performed to identify spatio-temporal overlaps, constructing the conflict set $\mathcal{C}$ as defined in Section 3.5.
4. Hamiltonian Mapping: The classical instance data is translated to construct the final cost Hamiltonian, which is subsequently optimized with a quantum computer.

4.3. QAOA and Subspace Restriction

The Quantum Approximate Optimization Algorithm (QAOA) [Farhi et al. 2014] is a hybrid classical-quantum algorithm designed to find approximate solutions to combinatorial optimization problems. To apply QAOA, the objective function is encoded into a cost Hamiltonian H_C, which is diagonal in the computational basis.

The QUBO formulation derived in Section 3 can be systematically converted into the Ising Hamiltonian H_C. This is achieved by mapping the binary decision variables $x_{v,r} \in \{0, 1\}$ to the Pauli-Z operators $Z_{v,r}$ with eigenvalues $\{+1, -1\}$ using:

$$x_{v,r} = \frac{I - Z_{v,r}}{2}, \tag{9}$$

where I is the identity operator. By substituting this mapping into the total cost function, any standard QUBO problem can be translated into the required Hamiltonian format.

In the standard QAOA framework, the system is initialized in an equal superposition of all possible states, $|\psi_0\rangle = |+\rangle^{\otimes n}$, and evolves under the alternating application of two unitaries driven by H_C and a mixer Hamiltonian H_M. The default mixer is typically the transverse field mixer, defined as $H_M = \sum_i X_i$, where X_i is the Pauli-X operator.

The parameterized quantum state is given by:

$$|\gamma, \beta\rangle = \prod_{k=1}^{p} e^{-i\beta_k H_M} e^{-i\gamma_k H_C} |\psi_0\rangle \,, \tag{10}$$

where p is the depth of the circuit, and γ and β are parameters optimized classically to minimize the expectation value $\langle\gamma, \beta| H_C |\gamma, \beta\rangle$.

However, the standard transverse field mixer explores the entire $2^{|V||R|}$ Hilbert space, which includes a vast number of invalid route assignments that violate our one-hot constraint. As one of the primary contributions of this work, we propose a tailored QAOA ansatz, inspired by the Quantum Alternating Operator Ansatz framework [Hadfield et al. 2019], designed to strictly restrict the search space to valid vehicle assignments.

Instead of relying solely on the quadratic penalty term H_{route} to enforce the condition softly, we restrict the exploration space by utilizing an XY-mixer alongside a custom initial state [Wang et al. 2020]. The proposed initial state $|\psi_{\text{init}}\rangle$ is constructed as the tensor product of valid one-hot states for each vehicle:

$$|\psi_{\text{init}}\rangle = \bigotimes_{v \in V} |W_{|R|}\rangle_v \,, \tag{11}$$

where $|W_{|R|}\rangle$ represents an equal superposition of all states with a Hamming weight of exactly 1 (*i.e.*, exactly one route selected per vehicle).

To ensure the system remains within this valid subspace during the quantum evolution, we replace the standard transverse field mixer with an XY-mixer. For each vehicle, the mixer is applied to its candidate routes arranged in a one-dimensional ring graph topology. By indexing the $|R|$ candidate routes of a vehicle v as integers from 0 to $|R| - 1$, we define the mixer Hamiltonian as:

$$H_{XY} = \frac{1}{2} \sum_{v \in V} \sum_{k=0}^{|R|-1} \left(X_{v,k} X_{v,(k+1) \bmod |R|} + Y_{v,k} Y_{v,(k+1) \bmod |R|} \right) , \tag{12}$$

where the modulo operation enforces the cyclic boundary conditions of the ring graph.

Because the XY-mixer preserves the Hamming weight of the state, the algorithm strictly explores the subspace of valid route assignments. This architectural modification drastically reduces the size of the search space. Furthermore, since the one-hot requirement is satisfied by design, the route constraint penalty term H_{route} can be entirely removed from the cost Hamiltonian, simplifying the final objective function:

$$H_C = \sum_{v \in V} \sum_{r \in R_v} c_{v,r} x_{v,r} + B \sum_{((v,r),(u,s)) \in \mathcal{C}} x_{v,r} x_{u,s}. \tag{13}$$

Consequently, the proposed parameterized quantum circuit for a given depth p is constructed by alternately applying the time-evolution operators associated with the simplified cost Hamiltonian H_C and the XY-mixer H_{XY} onto the valid-subspace initial state. The final QAOA ansatz is expressed as:

$$|\psi(\boldsymbol{\gamma}, \boldsymbol{\beta})\rangle = \prod_{k=1}^{p} e^{-i\beta_k H_{XY}} e^{-i\gamma_k H_C} |\psi_{\text{init}}\rangle \,. \tag{14}$$

5. Experimental Evaluation

To validate the proposed framework and compare the performance of the standard QAOA against our specialized QAOA Ansatz with the XY-mixer, we designed a controlled case study. This section details the experimental setup, the quantum execution parameters, and the measurement results.

5.1. Environment and Vehicle Instances

The simulation environment is defined as a 2D discrete lattice of dimension 21×21. Within this structure, we define a fleet consisting of two vehicles, $V = \{v_0,\ v_1\}$. Each vehicle $v \in V$ is assigned a specific terminal pair $(s_v,\ g_v)$ representing its start and end nodes, respectively:

- **Vehicle 0**: $s_0 = (0,\ 0), g_0 = (20,\ 20)$
- **Vehicle 1**: $s_1 = (20,\ 0), g_1 = (0,\ 20)$

Both terminal pairs are strategically positioned at the boundaries of the grid. Consequently, the optimal, shortest-path routes for both vehicles (*i.e.*, the main diagonal and anti-diagonal) will inevitably cross, triggering a collision. This layout forces the quantum solver to explore the trade-off between path efficiency and collision avoidance.

5.2. Candidate Routes

We utilize a fixed set of $|R| = 3$ candidate routes for each vehicle. While the framework can integrate with automated path-finding algorithms, this curated selection ensures the problem instance contains specific, unavoidable conflicts, providing a clear benchmark for the quantum solvers. The routes are defined as ordered sequences of lattice coordinates. The primary route for each vehicle ($r_{v,0}$) follows the shortest path, while secondary routes ($r_{v,1},\ r_{v,2}$) introduce localized deviations to explore the solution space. The resulting grid containing the vehicles and their respective routes is presented in Figure 2.

Vehicle 0: $(0,\ 0) \to (20,\ 20)$ The candidate set R_0 consists of the following routes:

Diagonal ($r_{0,0}$): $\{(i,\ i)\}_{i=0}^{20}$

Lower Deviation ($r_{0,1}$): $\{(0,\ 0)\} \cup \{(i+1,\ i)\}_{i=0}^{19} \cup \{(20, 20)\}$

Upper Deviation ($r_{0,2}$): $\{(0,\ 0),\ (0,\ 1)\} \cup \{(i,\ i+2)\}_{i=0}^{18} \cup \{(19,\ 20),\ (20,\ 20)\}$

Vehicle 1: $(20,\ 0) \to (0,\ 20)$ The candidate set R_1 consists of the following routes:

Anti-diagonal ($r_{1,0}$): $\{(20-i,\ i)\}_{i=0}^{20}$

Shifted Variation ($r_{1,1}$): $\{(20,\ 0)\} \cup \{(20-i,\ i+1)\}_{i=0}^{6} \cup \{(13,\ 8),\ (12,\ 8)\} \cup \{(20-i,\ i-1)\}_{i=9}^{20} \cup \{(0,\ 20)\}$

Complex Variation ($r_{1,2}$): $\{(20,\ 0)\} \cup \{(19-i,\ i)\}_{i=0}^{7} \cup \{(11-i,\ 7+i)\}_{i=0}^{10} \cup \{(1,\ 18),\ (0,\ 19),\ (0,\ 20)\}$

In this scenario, the temporal tolerance for collisions was set to $\Delta = 2$, implying that any pair of routes that contains the same node within two time units are identified as a conflicting pair. The penalty parameters related to the QUBO constraints A and B were scaled dynamically, corresponding to 1.5 times the length of the longest route in the instance.

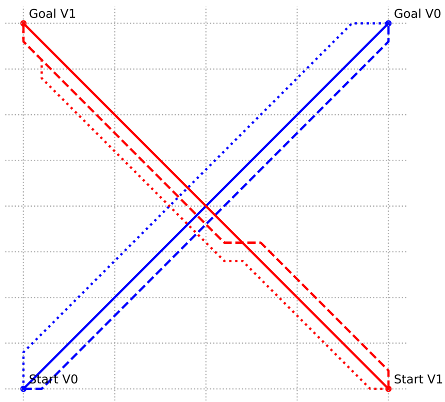


Figure 2. Experimental setup on a 21×21 grid showing candidate routes for two vehicles. Solid lines indicate the primary diagonal routes ($r_{0,0}, r_{1,0}$), which intersect and create a central conflict, while dashed and dotted lines show alternative routes in R_v that enable collision-free solutions.

Table 1. Hyperparameters and simulation settings for QAOA variants.

Parameter	Value
Number of Layers (p)	7
Classical Optimizer	COBYLA
Maximum Iterations	1,000
Final Measurement Shots	100,000

5.3. Quantum Execution and Simulation Parameters

Following classical pre-processing, the total Hamiltonian H was constructed. The implementation and quantum simulations were conducted using the Ket quantum programming platform [Da Rosa and De Santiago 2022].

To establish a classical benchmark, the Hamiltonian was first validated using a Simulated Annealing (SA) solver. The SA solver identified the ground state bitstring as 010100. Given the binary encoding $x_{v,r}$, where the first three bits represent the routes for v_0 and the subsequent three bits represent v_1, the result 010100 indicates the activation of $x_{0,1}$ and $x_{1,0}$. Physically, vehicle v_0 is assigned to its secondary route $r_{0,1}$ and vehicle v_1 follows its primary path $r_{1,0}$, avoiding the central collision while minimizing travel cost.

We evaluated both QAOA variants under two conditions: a noiseless exact simulation and a realistic noisy simulation. To initialize the parameters, we applied a strategy that emulates adiabatic evolution, setting $\gamma_k = \frac{k}{p}\Delta t$ and $\beta_k = (1-\frac{k}{p})\Delta t$, where $\Delta t = 0.5$. Table 1 summarizes the remaining hyperparameters used for all variational executions.

Noiseless Simulation The variational optimization loop utilized exact expectation value calculations natively provided by Ket. Final measurement outcomes were obtained by sampling the optimized state.

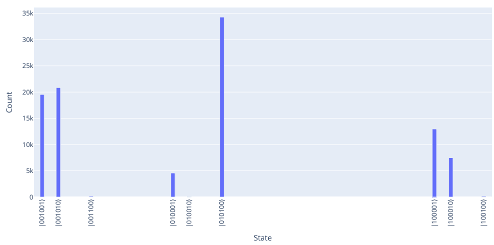
Noisy Simulation We utilized Ket's integration with Qiskit, employing the `Brisbane` QPU noise model with default settings. During the optimization loop, the expectation values were calculated using the Qiskit `Estimator` primitive, and the final measurement distributions were acquired using the `Sampler` primitive.

5.4. Results

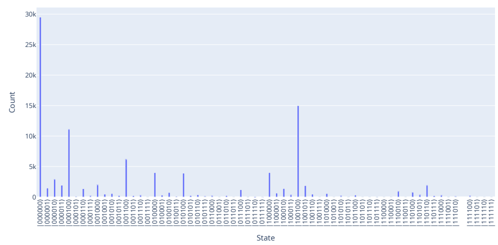
The measurement outcome distributions for all four experimental conditions are shown in Figure 3, utilizing 100,000 shots per final sampling. The optimal solution $|010100\rangle$ (cost 43, no collisions) serves as the reference state.

Under noiseless simulation, the QAOA Ansatz exhibits strong convergence, assigning 34,291 out of 100,000 samples ($\sim$34%) to the optimal state. The remaining probability mass is highly concentrated over other low-energy feasible states; for instance, the second most frequent state, $|001010\rangle$, reaches 20,847 counts at an energy of 46.00 (Figure 3a). Conversely, the standard QAOA under noiseless simulation fails to concentrate probability mass around the optimal state, ranking it in 7th position with only 3,906 counts ($\sim$3.9%). Its distribution is instead dominated by infeasible solutions, with the most frequent state being $|000000\rangle$ (29,513 counts) (Figure 3b). No significant concentration around low-energy feasible states is observed for the standard approach.

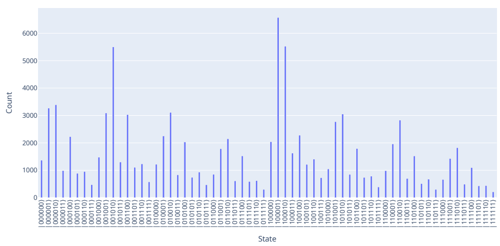
Under the IBM `Brisbane` QPU noise model, relying solely on peak probability is misleading. Although the standard QAOA measures the exact optimal solution more frequently (4,783 counts, $\sim 4.8\%$) than the QAOA Ansatz (2,035 counts, $\sim 2.0\%$), its distribution remains heavily dominated by unfeasible, high-energy states. This slight peak is attributed to a random flattening of the distribution caused by noise, rather than true algorithmic convergence (Figure 3d).In contrast, the QAOA Ansatz demonstrates superior resilience. While the XY-mixer's strict constraint-preserving properties degrade under noise, the algorithm successfully shifts the overall probability mass toward the low-energy subspace. As a result, its most frequent measurements remain low-energy, feasible states (Figure 3c), proving it achieves a lower expected energy and a significantly higher probability of sampling near-optimal solutions compared to the standard approach.



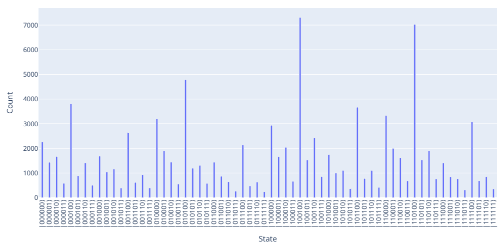
(a) QAOA Ansatz, noiseless.



(b) Standard QAOA, noiseless.



(c) QAOA Ansatz, IBM noise model.



(d) Standard QAOA, IBM noise model.

Figure 3. Measurement outcome distributions for all experimental conditions.

6. Algorithm Analysis and Scalability

The significant contrast in performance between the standard QAOA and the proposed QAOA Ansatz can be theoretically explained by analyzing the mathematical topology of the search space and the constraint mechanisms of both algorithms.

6.1. Search Space Pruning

In the standard QAOA implementation, the algorithm relies on the transverse-field mixer, which forces the system to navigate the entire Hilbert space defined by the n qubits. For an instance with $|V|$ vehicles and $|R|$ routes, the solver must explore $2^{|V|\cdot|R|}$ different computational states. For our specific experimental setup, this equates to $2^{2\cdot 3} = 64$ states.

By contrast, the proposed XY-mixer variant restricts the quantum evolution exclusively to the subspace where the one-hot constraint is strictly satisfied for each vehicle. Because each vehicle independently selects exactly one out of $|R|$ routes, the feasible search space is reduced to $|R|^{|V|}$. For our setup, this is $3^2 = 9$ states.

This represents an immediate reduction of approximately 86% in the state space for this specific, small-scale instance. However, as the problem scales to accommodate realistic drone fleets, the gap between the two approaches grows exponentially. For example, a scenario with 10 vehicles and 5 routes each results in a standard search space of 2^{50} states, whereas the Ansatz restricts the search to 5^{10} states, a reduction of over 99.99%. This exponential pruning of non-feasible configurations is critical for maintaining a high probability of success in real-world routing scenarios.

6.2. Hamiltonian Simplification and Convergence

Beyond mere size reduction, the QAOA Ansatz alters the optimization landscape. In the standard approach, the optimizer must balance the route selection constraint (H_{route}) against the cost minimization and collision penalties. If the penalty coefficient A is not tuned, the optimizer may fall into local minima where vehicles are assigned zero routes or multiple routes (as evidenced by the high probability of the infeasible state $|000010\rangle$ in our standard QAOA results).

By shifting the constraint enforcement from the cost Hamiltonian into the structure of the initial state and the mixer operator, we entirely eliminate H_{route}. This mathematical simplification ensures that the classical optimizer spends its limited iteration budget solely on minimizing path costs and avoiding collisions, rather than wasting resources learning basic assignment rules. Consequently, the Ansatz demonstrates significantly deeper concentration on low-energy states, proving to be fundamentally more scalable as problem dimensionality increases.

7. Final Remarks

This paper investigated the application of quantum optimization techniques to the drone collision avoidance problem. We introduced a QUBO formulation that captures route selection and collision constraints for multiple vehicles and solved it using QAOA. To improve performance, we proposed a tailored QAOA ansatz incorporating an XY-mixer that preserves the one-hot route assignment constraint and restricts the search space to feasible solutions. Experimental results under both noiseless and noisy simulations indicate that this constrained ansatz increases the probability of obtaining optimal or near-optimal solutions compared to the standard QAOA formulation. While noise in current quantum hardware still limits reliability, our results suggest that embedding problem constraints directly into circuit design can improve optimization performance and that hybrid classical-quantum approaches remain a promising direction for combinatorial problems such as UAV collision avoidance. Future work will investigate larger multi-vehicle scenarios, alternative mixer constructions, and tighter integration with classical routing heuristics.

Acknowledgments

AI tools were used only for language refinement. This work was partially supported by FAPESC (No. 2024TR002672) and INCT-CQA (CNPq, No. 408884/2024-0).

References

Clark, J., West, T., Zammit, J., Guo, X., Mason, L., and Russell, D. (2019). Towards real time multi-robot routing using quantum computing technologies. In *Proc. Int. Conf. on High Performance Computing in Asia-Pacific Region*, HPCAsia '19. ACM.

Da Rosa, E. C. R. and De Santiago, R. (2022). Ket quantum programming. *ACM Journal on Emerging Technologies in Computing Systems*, 18(1):1–25.

de Oliveira, F. M. C., Bittencourt, L. F., Bianchi, R. A. C., and Kamienski, C. A. (2024). Drones in the big city: Autonomous collision avoidance for aerial delivery services. *IEEE Transactions on Intelligent Transportation Systems*, 25(5):4657–4674.

Farhi, E., Goldstone, J., and Gutmann, S. (2014). A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*.

Haba, R., Mano, T., Ueda, R., Ebe, G., Takeda, K., Terabe, M., and Ohzeki, M. (2025). Routing and scheduling optimization for urban air mobility fleet management using quantum annealing. *Scientific Reports*, 15(1):4326.

Hadfield, S., Wang, Z., O'Gorman, B., Rieffel, E. G., Venturelli, D., and Biswas, R. (2019). From the quantum approximate optimization algorithm to a quantum alternating operator ansatz. *Algorithms*, 12(2):34.

Huang, Z., Li, Q., Zhao, J., and Song, M. (2022). Variational quantum algorithm applied to collision avoidance of unmanned aerial vehicles. *Entropy*, 24(11):1685.

Rezaee, M. R., Hamid, N. A. W. A., Hussin, M., and Zukarnain, Z. A. (2024). Comprehensive review of drones collision avoidance schemes: Challenges and open issues. *IEEE Transactions on Intelligent Transportation Systems*, 25(7):6397–6426.

Tang, J., Lao, S., and Wan, Y. (2022). Systematic review of collision-avoidance approaches for unmanned aerial vehicles. *IEEE Systems Journal*, 16(3):4356–4367.

Valavanis, K. P. and Vachtsevanos, G. J. (2014). *Handbook of unmanned aerial vehicles*. Springer Publishing Company, Incorporated.

Vianna, M. E. W. M., Panisson, A. R., and Rodrigues-Filho, R. (2025). Quantum computing for drone collision detection and avoidance: A systematic mapping study. In *VIII* $\langle WECIQ|WCQ\rangle$, Florianópolis, SC, Brasil. UFSC.

Wang, Z., Rubin, N. C., Dominy, J. M., and Rieffel, E. G. (2020). XY-mixers: analytical and numerical results for QAOA. *Physical Review A*, 101(1):012320.