



In-flight fast conflict-free trajectory re-planning considering UAV position uncertainty and energy consumption

Yutong Chen ^{b,c}, Yan Xu ^{a,b,*}, Lei Yang ^c, Minghua Hu ^c

^a Beihang University, Beijing, 100191, China

^b Cranfield University, Bedfordshire, MK43 0AL, United Kingdom

^c Nanjing University of Aeronautics and Astronautics, Nanjing, 210000, China



ARTICLE INFO

Keywords:

Unmanned aircraft system traffic management
Conflict detection and resolution
In-flight trajectory re-planning
Uncertainty
Energy consumption

ABSTRACT

This paper proposes a UAV in-flight fast conflict-free trajectory re-planning approach considering uncertainty and flight cost, which operates as an intermediate step between pre-flight conflict-free trajectory planning and real-time collision avoidance. It rapidly iterates the planned trajectory based on the dynamic environment during flight, aiming to balance optimality and robustness. Specifically, a probability-based grid occupancy model is introduced to upgrade conventional grid occupancy models to become compatible with conflict detection under uncertainty. A conflict-free trajectory search approach based on spatio-temporal reachable graphs is presented, which achieves rapid and conflict-free trajectory planning while considering controlled time of arrival (CTA) constraints and offering flexibility in selecting exit points within the airspace units. Experimental results demonstrate that the proposed method outperforms conventional full occupancy grid models and single-point rerouting models. In a representative experimental scenario with a $400 \times 400 \text{ m}^2$ single-flight-level airspace, the proposed method achieves a 100% success rate in conflict resolution at a traffic density of up to 40 UAVs per minute. The average computing time per UAV is approximately 0.04 s, and the average extra energy consumption rate per adjusted UAV is only about 0.55%.

1. Introduction

With the rapid development of the unmanned aerial vehicle (UAV) industry, UAVs are increasingly being utilised in various aspects of production and daily life (SJU, 2018), including logistics (Li et al., 2022), rescue operations (Silvagni et al., 2017), agricultural pest control (Freitas et al., 2020), inspections (Wu et al., 2021), and more. According to reports, by 2030, within the United Kingdom alone, there are expected to be approximately 900,000 UAVs operating in the skies, with an estimated economic impact reaching £45 billion (PwC, 2018). UAVs primarily operate within low-altitude airspace, specifically below 400 ft, necessitating research in unmanned aircraft system traffic management (UTM) for low-altitude airspace (FAA, 2023). However, due to the complex nature of UAV operations compared to commercial aviation and the fact that UTM-related operational technologies are still being developed (Pang et al., 2022), there are currently limited scenarios for large-scale UAV operations today. In order to meet the need of supporting the growing demand for UAV operations in low-altitude airspace, conflict detection and resolution (CD&R), as a key UTM technology, has been widely studied to tackle the high-density complex scenarios, Kleinbekman et al. (2020), aiming to achieve safe and efficient low-altitude airspace operations in a stable and rapid way.

* Corresponding author at: Beihang University, Beijing, 100191, China.

E-mail addresses: yutong.chen@cranfield.ac.uk (Y. Chen), yan.xu@buaa.edu.cn (Y. Xu), laneyoung@nuaa.edu.cn (L. Yang), minghuahu@nuaa.edu.cn (M. Hu).

In terms of control methods, UAV CD&R methods are typically classified into centralised and distributed approaches.

- Centralised methods usually involve considering all relevant UAVs and constructing a global optimisation problem, such as using 0-1 linear programming (Yang et al., 2019), mixed-integer linear programming (Tan et al., 2019), mixed-integer nonlinear programming (Zhang et al., 2018), mixed-integer quadratic programming (Mellinger et al., 2012) and others, assuming access to global information. The advantage of this approach is that, based on a global optimisation model, solvers can theoretically find the optimal solution. However, the challenge with this approach arises when the problem becomes too large or involves a significant number of factors, which may result in an exceedingly large model, making it difficult for solvers to find the optimal solution within a reasonable time frame. Therefore, although there are several exact methods (i.e., branch-and-cut algorithm (Cheng et al., 2020) and dynamic programming (Peng et al., 2021)) are used to solve the global optimisation problem, approximate methods like genetic algorithms (Roberge et al., 2012), auction algorithm (Yao et al., 2019), simulated annealing (Ait-Saadi et al., 2022), particle swarm optimisation (Hong et al., 2017), and wolf pack algorithms (Chen et al., 2018) are often employed to find an acceptable solution in a reasonable amount of time. Nevertheless, this approach falls short of achieving the vision of obtaining a globally optimal solution through centralised methods, and the stability of solutions obtained by these intelligent algorithms is difficult to prove (Wu, 2021). Moreover, every time the environment changes, the model needs to be rebuilt and solved, which is often time-consuming, making centralised methods challenging for addressing dynamic environments (uncertainties).
- Distributed methods, on the other hand, typically create separate subproblem models for each UAV or potential conflict and solve these subproblems individually to address the global CD&R issue. In a distributed control system, the individual entities (such as UAVs) can be cooperative (Ning et al., 2019) or non-cooperative (Yang et al., 2013). Distributed methods are generally less sensitive to problem scale and exhibit greater robustness (Yang and Wei, 2020). While distributed methods may not have the theoretical potential to find a globally optimal solution due to the loss of global information, they still hold the promise of improving solution optimality locally, for example, by constructing planning models for subproblems. Deploying agents is a commonly used technique in distributed methods. To better cope with environmental uncertainties, these agents often control UAV motion continuously based on a set period or iterative refine previous trajectories through methods like distributed model predictive control (MPC) (Luis et al., 2020), A-star (Penin et al., 2018), Markov decision processes (MDP) (Ong and Kochenderfer, 2017), deep reinforcement learning (DRL) (Brittain and Wei, 2022; Zhang et al., 2023), and more.

In terms of the operational phase, methods for UAV CD&R are typically divided into pre-flight and in-flight CD&R.

- Pre-flight CD&R aims to plan conflict-free trajectories based on airspace operational data (including current and planned data) before UAV takeoff, reducing the potential for conflicts during flight. Therefore, it is also referred to as pre-flight conflict-free trajectory planning. The pre-flight CD&R often has less stringent computational time requirements for solvers, allowing for larger-scale model computations. Therefore, besides safety, pre-flight trajectory planning can consider more objectives related to flight costs, such as flight distance, delay time, energy consumption, etc. Adjustments can be made not only to speed (Seo et al., 2017) and altitude (Tang et al., 2021b) but also through rerouting (Wilhelm and Clem, 2019) to fine-tune trajectories and achieve the reachability of specific waypoints. It should be noted that although it is possible to plan conflict-free trajectories for all UAVs before takeoff, it is still challenging to avoid potential conflicts during actual flights. This is due to uncertainties in flight, making it difficult for all UAVs to precisely follow the pre-flight planned conflict-free trajectories in both time and space. In the context of pre-flight trajectory planning for commercial aviation, uncertainties, such as weather (Courchelle et al., 2019) and trajectory prediction (Huang et al., 2021), are widely considered. However, for UAVs, due to their smaller size compared to commercial flights and their more flexible and dynamic operations, uncertainties for CD&R are typically addressed not before takeoff but during flight (Yang and Wei, 2021). This, to some extent, may reduce the robustness of pre-flight trajectory planning.
- In-flight CD&R, on the other hand, aim to detect potential conflicts in real-time during UAV flight and respond quickly to resolve them (González-Arribas et al., 2018). Since the need for in-flight conflict resolution is more immediate, uncertainties are typically considered to enhance the solver's practical applicability (Yang and Wei, 2020). However, as this CD&R process must occur during flight, it places higher demands on the solver's computation time. This limitation restricts the model's scale, resulting in in-flight CD&R methods primarily focusing on safety objectives and giving less consideration to cost-relevant objectives (including energy consumption). Existing in-flight CD&R methods for UAVs are similar to the traffic alert and collision avoidance system (TCAS) (Livadas et al., 2000) or the next-generation airborne collision avoidance system (ACAS) (Kochenderfer et al., 2012) used in commercial aviation, which is responsible for terminal collision avoidance. It should be noted that there are typically two approaches in in-flight CD&R: one is to conduct conflict-free trajectory planning in advance, and the other is to make dynamic continuous decisions and manoeuvres; in this paper, collision avoidance refers to the latter. To ensure the safe, stable, and rapid resolution of potential conflicts, in-flight CD&R methods prefer to use rapid and efficient approaches such as optimal reciprocal collision avoidance (ORCA) (Ho et al., 2018), rapidly-exploring random trees (RRT) (Zammit and Van Kampen, 2022), RRT* (Guo et al., 2022), and fast Markov decision process (FastMDP) (Bertram et al., 2022) within a distributed control framework. However, as the considered environment becomes more complex, the performance of these methods may significantly deteriorate (Bertram et al., 2023). While reinforcement learning methods have been explored for in-flight CD&R in complex scenarios (Wang et al., 2022), their stability still requires further validation.

Table 1

Layered CD&R architecture and characteristics of each method.

Module	Pre-flight conflict-free trajectory planning	In-flight conflict-free trajectory re-planning ^a	Collision avoidance
Space range	Global airspace	Airspace unit	Local airspace
Time range	Before take-off	Before entering the airspace unit	Before collision
Advantage	Optimality	Balance optimality and robustness	Robustness

^a The proposed method.

Since UAVs are generally smaller in size and lighter in weight compared to traditional commercial aircraft, they are more easily affected by environmental factors, making it crucial to account for flight uncertainties. A chance-constrained optimisation with fast disruption recovery under uncertain waypoint occupancy time is proposed to address UTM traffic flow issues (Pang et al., 2024). The Monte Carlo method is effective in estimating conflicts between aircraft but is computationally intensive, making it inefficient (Yang et al., 2020). A chance-constrained programming model considers various uncertainties for conflict avoidance but can be computationally expensive, especially as the configuration space dimensions increase (Du et al., 2021). A method for planning a trajectory for a single vehicle involves converting probabilistic constraints to deterministic ones (Chai et al., 2020). A customised decentralised online algorithm based on Monte Carlo Tree Search significantly enhances computational efficiency but still faces challenges in real-time computing scenarios (Wu et al., 2022b). Basically, when considering uncertainties in UAV optimisation problems, the complexity of the problem significantly increases, thereby extending computation time. Compared to traffic flow management issues during the pre-tactical phase, the in-flight CD&R problems during the tactical phase are more sensitive to computation time. Therefore, there remains a need to explore trajectory planning approaches considering uncertainties that better meet real-time computational requirements, such as grid-based methods (Dai et al., 2021) that trade storage space for computation time.

Based on the discussion above, pre-flight conflict-free trajectory planning is faced with challenges in dealing with uncertainties (i.e., poor robustness), which can lead to potential conflicts during the flight. Collision avoidance also poses challenges in achieving cost-relevant objectives. Therefore, this paper proposes a method for in-flight conflict-free trajectory re-planning tailored to high-density complex scenarios, taking into account uncertainties and flight costs. The space range and time range of the proposed method both fall between that of pre-flight conflict-free trajectory planning and collision avoidance. We envision a layered architecture for conflict resolution, which comprises the following three modules (as shown in Table 1): (a) pre-flight conflict-free trajectory planning, (b) the proposed in-flight conflict-free trajectory re-planning, and (c) collision avoidance (if needed). The proposed method is expected to, on the one hand, better address uncertainties by obtaining a more accurate assessment of the dynamic environment during the flight compared to that of pre-flight trajectory planning. On the other hand, it can produce optimised rerouting trajectories in terms of flight costs than following collision avoidance actions.

The main contributions of this paper are as follows:

1. A probability-based grid occupancy model is introduced to enable the airspace grid model to accommodate the uncertainty of UAV positioning error, which is assumed to be subject to a two-dimensional Gaussian distribution.
2. A conflict detection method based on the spatio-temporal grid reflecting occupancy rate is proposed to facilitate probabilistic conflict identification. It also ensures that the detection time does not significantly increase with the number of UAVs of interest. A grid occupancy rate iterative update method and an occupying rate distribution nominal map are employed to further enhance computational speed.
3. A conflict-free trajectory search method based on reachable spatio-temporal graphs is proposed. This approach transforms the problem of conflict-free trajectory planning with controlled time of arrival (CTA) constraint into a directed graph-based path planning problem, enabling the rapid generation of conflict-free trajectories.
4. In the representative scenarios constructed in this paper, the proposed method demonstrates competitive performance in terms of effectiveness, efficiency, stability, and timeliness compared to conventional grid-based models and single-waypoint rerouting models.

The rest of this paper is organised as follows. We first describe the studied CD&R problem for UTM in Section 2. Then, Section 3 introduces the proposed CD&R method based on grid probability occupancy in detail. Next, simulation experiments are conducted in Section 4. After that, we discuss this study further in Section 5. Finally, Section 6 concludes the research findings.

2. Problem formulation

The conflict detection and resolution problem for UTM consists of conflict detection (CD) and conflict resolution (CR). CD determines whether there is a potential conflict based on the current trajectory. If there is a potential conflict, CR is triggered. This paper defines conflict resolution for UTM as the re-planning of conflict-free trajectories by selecting one or more rerouting points in the airspace unit, similar to the sectors in air traffic management (ATM). Suppose that UAVs pass through several airspace units in their flights. In that case, if they can quickly plan their conflict-free trajectories in an airspace unit before they are about to enter the unit, the global scope of the in-flight re-planning task can be realised. As shown in Fig. 1, a UAV passes successively through the airspace units U1, U2, U3, and U4. For example, when it is flying in U2 and about to enter U3, CD&R is carried out

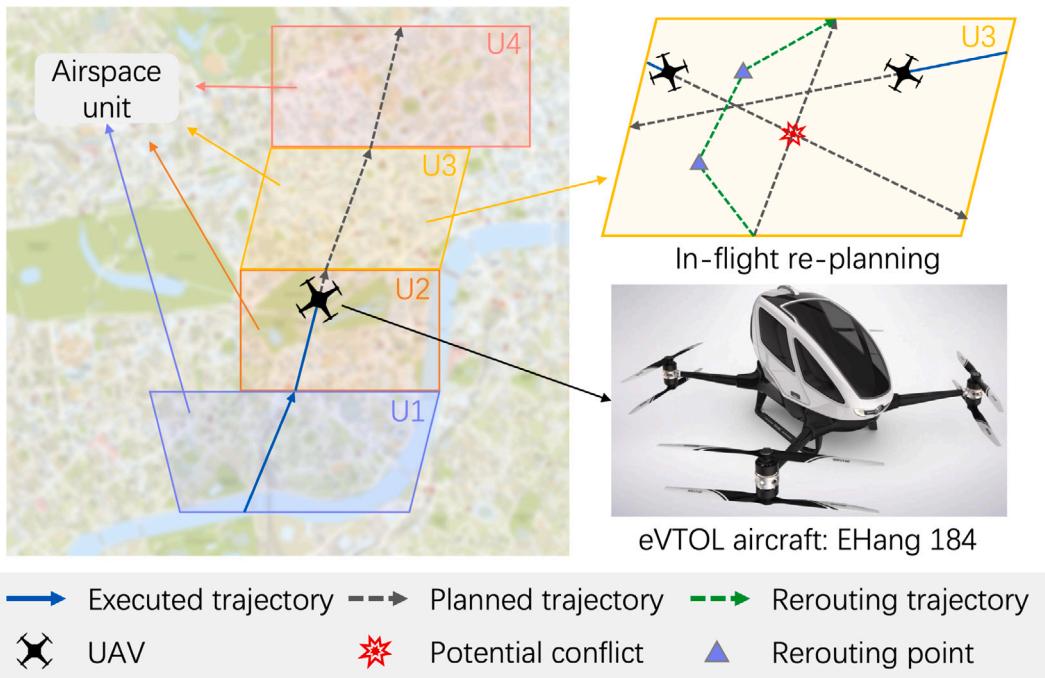


Fig. 1. Operation scenario of the proposed CD&R for UTM. The CD&R of each UAV is based on the airspace unit as the basic operating range and is executed in sequence according to the airspace unit it passes through.

based on current airspace operating information in U3. If a potential conflict is detected, the UAV's conflict-free trajectory in U3 is quickly generated. Under this approach, the scale of the individual problem being solved is primarily related to the size of the airspace units and is almost independent of the total distance of the entire flight. This effectively decomposes the global problem to facilitate its resolution. Due to the diverse range of UAVs, for the sake of convenience in research, this paper instantiates them with a commonly used electric vertical takeoff and landing (eVTOL) aircraft, the EHang 184. Nevertheless, we will discuss how the proposed approach is compatible with fixed-wing aircraft (refer to Section 5). For consistency, we will continue to use UAV to refer to this instance of eVTOL in the subsequent text. It should be noted that, for readability, the size of elements in the illustrative images in this paper (such as the UAVs and maps) may not reflect real-world proportions. We will provide detailed information on the parameters used in the experiments in Section 4.

This study assumes a possible operating environment, as shown in Fig. 2. There are four main stakeholders in this environment:

1. Common information service provider (CISP): It enables the exchange of essential information between the authorised stakeholders (EASA, 2020). This study focuses on its discovery & synchronisation service (DSS) function based on airspace occupancy.
2. UTM service provider (UTMSP): It provides flight-related services for its users, and this study focuses on in-flight trajectory re-planning services. UTMSP holds the operational information of the UAV it serves, such as flight task (e.g., times of arrival at the entry and exit points), aircraft performance and planned trajectory.
3. Ground control station (GCS): It controls UVA flights based on the service information provided by UTMSP.
4. UAV: It performs flight tasks based on the trajectory information received.

Other components of UTM (e.g., air navigation service provider and supplementary data service provider) are referred to the Open-Access UTM framework (Catapult, 2021) and will not be mainly discussed in this paper. This environment assumes that data link nodes include cloud servers (including but not limited to ground communication stations and 4G/5G networks), ground control stations, and UAVs.

The environment consists of three core functional components related to CD&R tasks:

1. Airspace occupancy (hosted by the CISP): It updates airspace occupancy in real-time to support CD&R.
2. Conflict detection (hosted by UTMSPs): It determines whether there is a potential conflict based on a UAV's current trajectory and airspace occupancy.
3. Conflict resolution (hosted by UTMSPs): It generates a conflict-free trajectory based on the UAV's flight task and airspace occupancy when a potential conflict is detected.

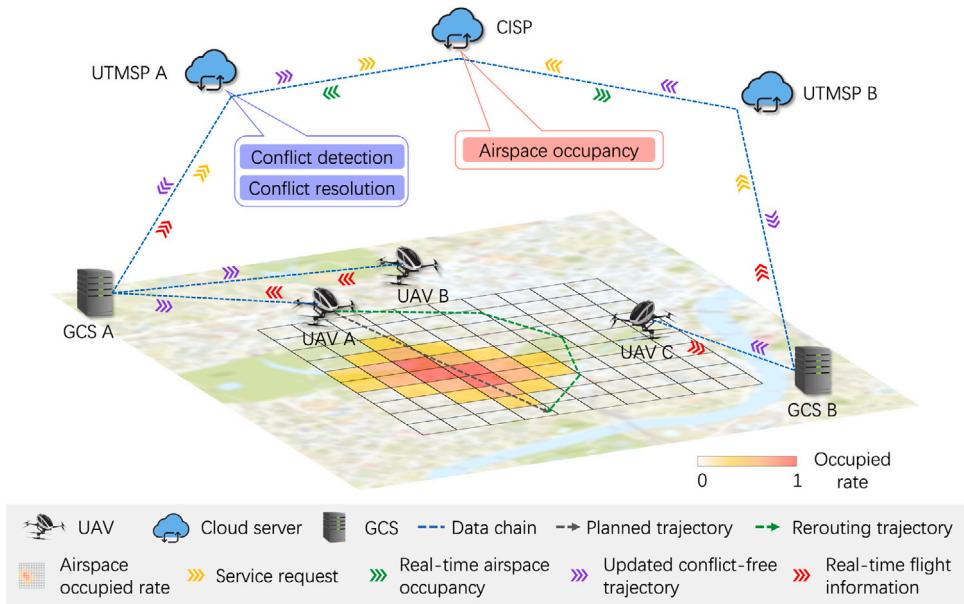


Fig. 2. Workflow of the proposed CD&R of UTM.

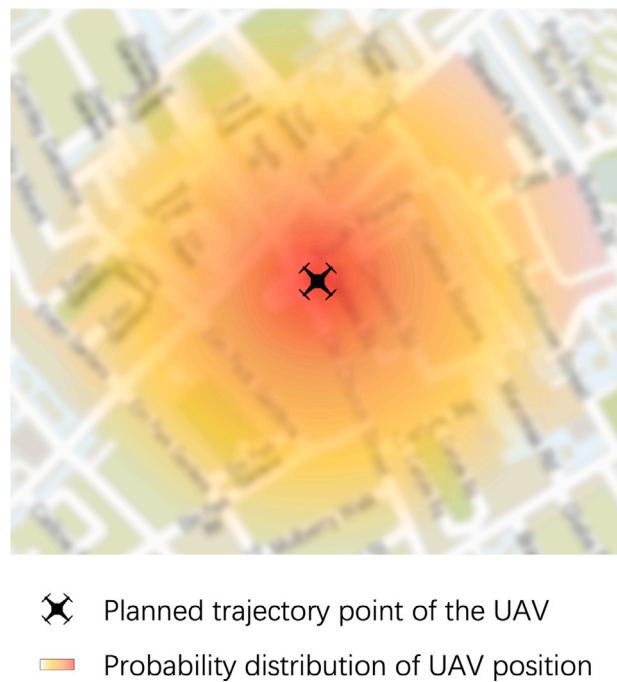
In Fig. 2, we assume that UAVs A and B are controlled by GCS A, which is served by UTMSP A, and UAV C is controlled by GCS B, which is served by UTMSP B. In this environment, the CD&R process for a UAV (taking UAV A as an example) is described as follows:

1. When GCS A identifies UAV A is about to enter the airspace by the real-time flight information, it sends a CD&R service request (refer to the yellow arrow marks) to its service provider, i.e., UTMSP A.
2. Upon receiving the CD&R service request from GCS A, UTMSP A immediately executes the conflict detection component, which determines whether there is a potential conflict based on real-time synchronised airspace occupancy and UAV A's current trajectory. The CISP distributes the updated airspace occupancy via its cloud server to the relevant UTMSPs' cloud servers (refer to the green arrow marks).
3. (a) If UAV A's current trajectory is not conflict-free, UTMSP A will execute the conflict resolution component based on real-time synchronised airspace occupancy and UAV A's flight task to generate a conflict-free trajectory; (b) otherwise, no operation is performed. Note that the following steps are based on situation (a).
4. UTMSP A sends the updated conflict-free trajectory to UAV A via GCS A for execution and to the CISP for updating airspace occupancy (refer to the purple arrow marks).
5. UAV A executes the updated conflict-free trajectory, and the CISP distributes the updated airspace occupancy to the relevant UTMSPs.

To clarify this study's CD&R problem for UTM further, the following notes are provided:

- Assumption

1. Only horizontal manoeuvres are considered.
2. The airspace unit is provided with several fixed entry and exit points, similar to the free route airspace (FRA) (Gaxiola et al., 2018) in ATM.
3. Communication delay and operation delay are not considered.
4. The kinematic state of an aircraft (e.g. speed and heading) changes instantaneously.
5. UAV departs from its pre-defined entry point and arrives at its pre-defined exit point or one of the alternative exit points near its pre-defined exit point.
6. Only the positioning error is considered as flight uncertainty and all UAVs have the same level of error.
7. Weather conditions are not considered.
8. The First-Come-First-Served (FCFS) rule is used, which is a common sequential decision strategy that is considered fair (Ho et al., 2019; Tang et al., 2021a). The UAV that enters the airspace unit first has a higher priority, and the UAV is responsible for ensuring safety between itself and UAVs with higher priority.



- ❖ Planned trajectory point of the UAV
- Probability distribution of UAV position

Fig. 3. Schematic diagram of UAV position distribution probability.

- Constraint

1. Flight safety: Flight safety performance must not exceed the preset minimum.
2. Aircraft performance: For UAVs, the trajectory must be executable within their performance envelope.
3. CTA: If feasible, the trajectory should meet the CTA requirement at the exit point.

- Objective

- Minimise energy consumption.

3. Methodology

In this section, we first establish a gridded airspace occupancy model designed to characterise the uncertainty of UAV positions (Section 3.1). Subsequently, we derive a UAV conflict detection model based on the proposed grid occupancy model (Section 3.2). Finally, we introduce an approach to transform the conflict resolution problem into a shortest path problem based on directed weighted graphs to rapidly obtain the optimal solution (Section 3.3.3).

3.1. Probability-based grid occupancy model

Airspace gridding is a common approach in UTM airspace modelling (Dai et al., 2021). Given that current UTM research often assumes the airspace to be compatible with free-flight (which is able to handle a higher air traffic density even in the presence of various uncertainties and delays and can also increase fuel and time efficiency (Valenti Clari et al., 2001)), the grid's centre points can be used as virtual waypoints for trajectory planning once the airspace is gridded (Wu et al., 2022a). Grids can also be assigned attributes such as availability, ground risk, weather conditions, etc., to represent the conditions within their encompassed area. Additionally, grids aid in conflict detection (Perez-Leon et al., 2021). In existing methods, it is usually assumed that an aircraft occupies the grid centred around its position or a few adjacent grids. It is identified as a conflict if UAVs occupy overlapping grids in space and time. To accommodate uncertainty, this paper introduces a probability-based grid occupancy model for the proposed method.

Since this study only considers positioning error in flight uncertainty, we assume that the position distribution of the UAV follows a two-dimensional Gaussian distribution, as shown in Fig. 3. In order to facilitate the use of the conflict detection model based on grid probability occupancy, this paper takes the integral of the probability of the position distribution of the UAV in a grid as the UAV's occupying rate, as shown in Fig. 4. Readers are referred to Appendices B and C for more details.

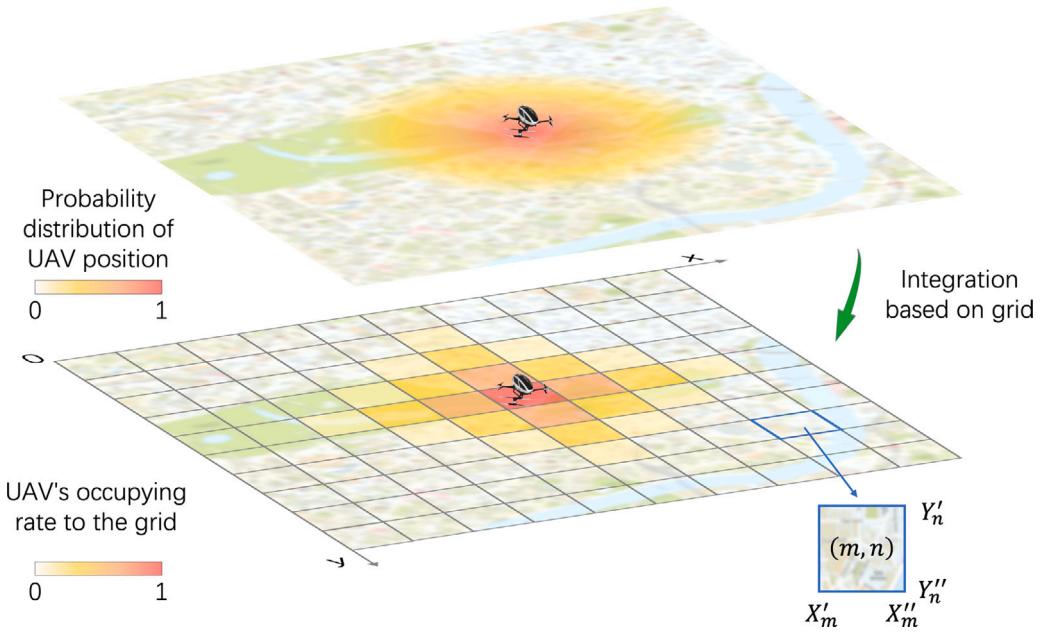


Fig. 4. Grid-based probability integral calculation of UAV position distribution.

3.2. Conflict detection based on the occupancy rate of spatio-temporal grid

There are three commonly used approaches for conflict detection with uncertainty:

1. The first approach involves random sampling of predicted trajectories with uncertainty to obtain a deterministic trajectory, followed by conflict judgement on these specific trajectories (Dias et al., 2022; Dias and Rey, 2022). This method is typically used in trajectory planning-based models, such as model predictive control. However, the application of this method presents significant challenges due to potential disparities between sampled trajectories and actual trajectories.
2. The second approach obtains probabilities through simulation-based Monte Carlo methods (Alejo et al., 2016; Jenie et al., 2017), which may face challenges in balancing computational accuracy and simulation time, especially in high-density and complex scenarios.
3. The third approach calculates the overlapping volume or area of the position distribution regions of two UAVs (Zou et al., 2021; Seyedipour et al., 2022). While this method effectively represents the probability of conflict occurrences, it necessitates pairwise calculations for all relevant aircraft, making it challenging to integrate with grid-based occupancy models and resulting in slow computation in high-traffic density scenarios.

Building upon the efficient uncertainty representation in approach 3, this paper attempts to further address the issue of slow computational speed in high-traffic density scenarios. Therefore, this paper proposes a conflict detection method based on spatio-temporal grid occupancy rates to extend the conventional entire occupancy model into a probability occupancy model of the grid, which aims to merge uncertainty and grid-based conflict detection. This approach circumvents the need for UAVs within the scenario to conduct pairwise conflict detection sequentially. Instead, it only requires a single conflict detection process based on UAVs' occupancy of the airspace grid. This ensures that the detection time does not increase with the growing number of UAVs in the scenario.

3.2.1. Spatio-temporal grid model

In order to facilitate conflict detection, we construct a spatio-temporal grid model, as shown in Fig. 5. The spatial dimension is represented by the plane xOy . The t -axis represents the time dimension, where the interval is ΔT and each moment is called a time step. The UAV is located in the centre of a grid at each time step. According to the probability-based grid occupancy model, we can get the UAV's occupying rate to each grid at each time step. We define grid (m, n) at time step τ as grid (m, n, τ) . Therefore, we use $p_i^{(m,n,\tau)}$ to represent UAV i 's occupying rate to grid (m, n, τ) (refer to Eq. (C.1)). The physical meaning of $p_i^{(m,n,\tau)}$ is the probability that UAV i will appear in grid (m, n, τ) .

3.2.2. Safety constraint

In this paper, it is considered as safe if the probability that two or more UAVs appear in a grid is less than or equal to threshold P^{Safety} . Then, the safety constraint for grid (m, n, τ) can be represented as:

$$p_{(\geq 2)}^{(m,n,\tau)} \leq P^{\text{Safety}} \quad (1)$$

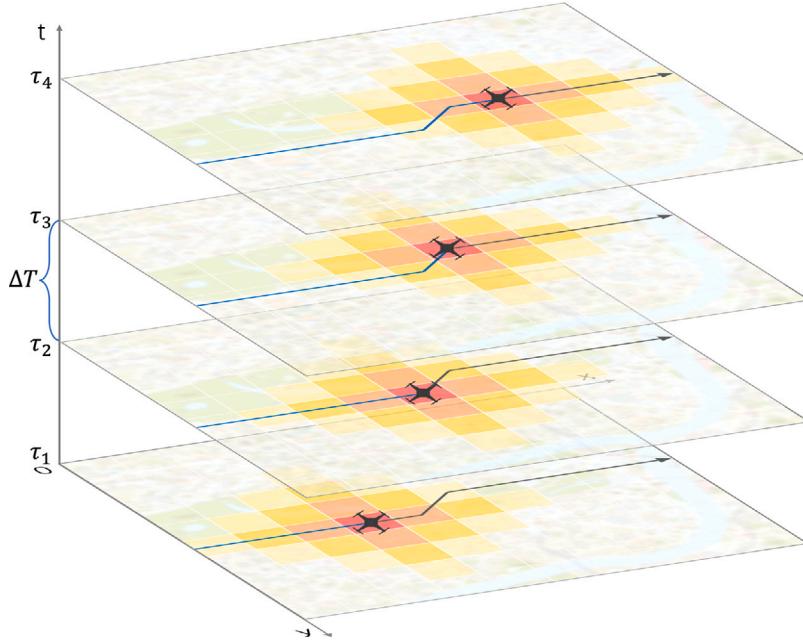


Fig. 5. Spatio-temporal grid representation of UAV position.

where $p_{(\geq 2)}^{(m,n,\tau)}$ represents the probability of no less than 2 UAVs in grid (m, n, τ) . Since the occupation of a grid by UAVs is independent of each other, $p_{(\geq 2)}^{(m,n,\tau)}$ can be represented as:

$$p_{(\geq 2)}^{(m,n,\tau)} = 1 - p_{(0)}^{(m,n,\tau)} - p_{(1)}^{(m,n,\tau)} \quad (2)$$

where $p_{(0)}^{(m,n,\tau)}$ and $p_{(1)}^{(m,n,\tau)}$ represent the probabilities of 0 and 1 UAV appearing in grid (m, n, τ) , respectively. $p_{(0)}^{(m,n,\tau)}$ and $p_{(1)}^{(m,n,\tau)}$ can, respectively, be represented as:

$$p_{(0)}^{(m,n,\tau)} = \prod_{i \in I^{(m,n,\tau)}} \left(1 - p_i^{(m,n,\tau)}\right) \quad (3)$$

$$p_{(1)}^{(m,n,\tau)} = \sum_{i \in I^{(m,n,\tau)}} \left[p_i^{(m,n,\tau)} \times \prod_{i' \in I^{(m,n,\tau)}}^{\neq i} \left(1 - p_{i'}^{(m,n,\tau)}\right) \right] \quad (4)$$

where $I^{(m,n,\tau)}$ represents the set of UAVs whose probability of occupying grid (m, n, τ) is greater than or equal to Φ . As mentioned in Section 3.1, we introduce the threshold Φ to facilitate calculation. Eq. (2) can be rewritten as:

$$p_{(\geq 2)}^{(m,n,\tau)} = 1 - \prod_{i \in I^{(m,n,\tau)}} \left(1 - p_i^{(m,n,\tau)}\right) - \sum_{i \in I^{(m,n,\tau)}} \left[p_i^{(m,n,\tau)} \times \prod_{i' \in I^{(m,n,\tau)}}^{\neq i} \left(1 - p_{i'}^{(m,n,\tau)}\right) \right] \quad (5)$$

Therefore, since this paper adopts the FCFS rule, conflict detection can be made by judging whether the grids' occupied rates still satisfy the safety constraint after the new UAV enters the airspace unit, as shown in Fig. 6. If they satisfy the safety constraint, the current planned trajectory of the new UAV is conflict-free.

It should be noted that, calculating Eq. (5) is complex and time-consuming, especially with a large number of UAVs in the airspace unit. Therefore, we use iterative calculation to solve this problem. That is, as long as the new UAV's occupying rate to grid (m, n, τ) , $p_{\text{New}}^{(m,n,\tau)}$, is less than or equals to the remaining occupiable rate $r^{(m,n,\tau)}$, the safety constraint will be satisfied. Readers are referred to Appendix D for more details.

3.3. Conflict-free trajectory search based on spatio-temporal reachable graph

As described in Section 2, in order to ensure the stability of global airspace operations, this paper introduces CTA constraints and FCFS rules. In this context, the trajectory planning problem in a grid-based airspace model is typically addressed by constructing a trajectory planning model and then solving it using exact or approximate algorithms (Tang and Xu, 2023). Since this paper focuses on conflict detection and resolution in high-density complex scenarios, exact algorithms face computational challenges in terms of time, while approximate algorithms cannot guarantee optimal solutions.

The new UAV's occupying rates to grids

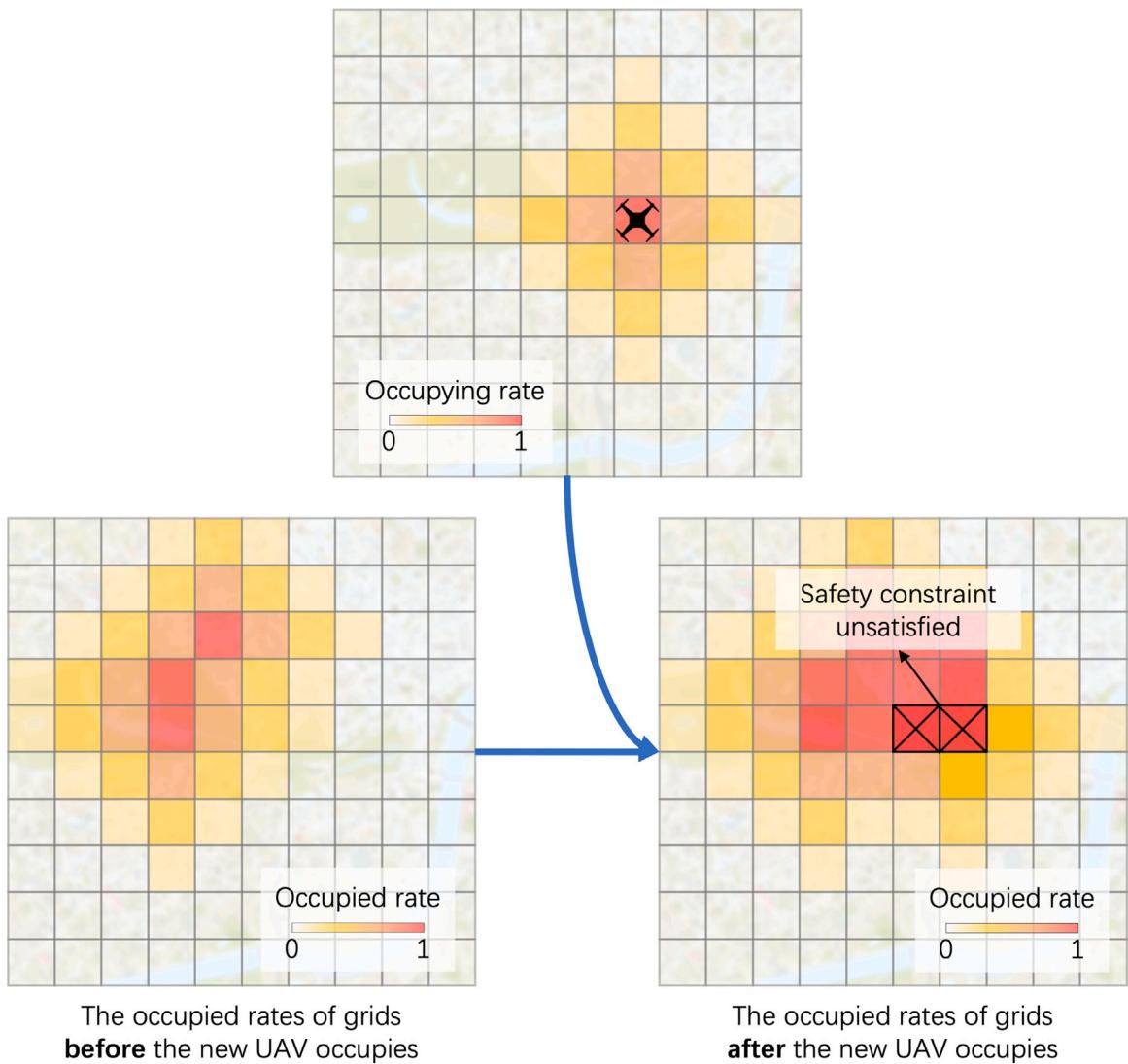


Fig. 6. Calculation process of conflict detection based on the occupancy rate of spatio-temporal grid.

Therefore, this paper transforms this conflict-free trajectory planning problem into a directed graph-based shortest path problem, allowing for rapid attainment of optimal solutions for each UAV in sequence. Graph theory has been widely applied to trajectory planning problems, such as constructing visibility graphs (Tang et al., 2021b) or Voronoi graphs (Hu et al., 2020). However, previous graph-based methods rarely consider CTA restrictions. To address this issue, this paper introduces a conflict-free trajectory search based on a spatio-temporal reachability graph, which not only accommodates CTA constraints but also provides a flexible exit point function.

As mentioned in Fig. 2, the conflict resolution component will be executed if the current planned trajectory is not conflict-free. In this paper, we choose a feasible rerouting point at each time step based on the spatio-temporal grid model. The chosen rerouting points and the UAV's entry and (alternative) exit points form a conflict-free rerouting trajectory. Due to the fact that UAVs typically rely on batteries as their power source, the amount of energy they can carry is limited by their small size. This leads to the critical importance of energy consumption in UAV trajectory planning (Cheng et al., 2020).

3.3.1. Feasible rerouting point

This paper considers all grids' centre points as potential rerouting points. To facilitate the description of how to find feasible rerouting points, we define three types of grids based on the spatio-temporal grid model:

1. Reachable grid: It refers to the grid that can be reached with the UAV performance. Reachable grids are divided into available grids and unavailable grids.
2. Available grid: The safety constraint is met when the UAV enters this grid.
3. Unavailable grid: The safety constraint is NOT met when the UAV enters this grid.

Therefore, to find feasible routing points is to find reachable grids, and the process for a time step (i.e., a two-dimensional plane of the spatio-temporal grid model) is shown in Fig. 7, which includes sub-figures (1)–(4) for the sake of illustration. First, as shown in sub-figure (1), this study uses space-time prism theory to identify reachable grids. Specifically, for a given plane, two circles are drawn, respectively, with the UAV's entry and exit points as centres. The radius of each circle is the product of the time from the point (entry/exit point) to the plane and the UAV's maximum speed. The overlapping area of these two circles forms the reachable area, and the grid, whose central point within this area, is identified as the reachable grid. Next, for a reachable grid, assume that the UAV is in the grid, based on the UAV's occupying rates to grids in this case (refer to sub-figure (2.1), (2.2) or (2.3)) and the current grids' occupied rates (refer to sub-figure (3)), and then determine whether the safety constraint is satisfied by the conflict detection method mentioned in Section 3.2 (especially refer to Fig. 6). If the safety constraint is satisfied, the reachable grid is an available grid; otherwise, it is an unavailable grid (e.g., sub-figure (4)). Finally, we obtain the set of candidate nodes used to generate the directed graph for UAV i , represented by $N_i^{\text{Candidate}}$. Readers are referred to Appendix E for more detailed information regarding the method for identifying feasible rerouting points.

3.3.2. Weighted directed graph

Fig. 8 illustrates the method for generating the directed graph and the optimal path. According to the edges of points of set $N_i^{\text{Candidate}}$ and from UAV i 's entry point to exit point, edges are generated layer by layer in time step order. The constraints required to generate an edge are as follows:

1. Connection constraint: The start point of the candidate edge must be the end point of an already generated edge in the previous time step unless it is UAV i 's entry point.
2. Speed constraint: The flight (i.e., distance and time) represented by the edge is within the speed performance of the UAV.
3. Reachability constraint: The number of time steps from the point at the back of the edge (i.e., the time step farther back) to the exit point meets the CTA requirements of the exit point.
4. Entrance constraint: The UAV shall not stay in the grid around entry points after entering the airspace unit so as to prevent the UAV that enters after it from facing an inextricable conflict as soon as it enters the airspace unit.
5. Direction constraint: The flight that the edge indicates is close to the exit point rather than away from it.

Based on the connection constraint, we use $E_{i,\tau}^{\text{Candidate}}$ to represent the set of candidate directed edges between the adjacent time step planes ($\tau - 1$) and τ , and it can be represented as:

$$E_{i,\tau}^{\text{Candidate}} = \left\{ e_{(m,n,\tau)}^{(m',n',\tau-1)} \middle| \begin{array}{l} g_{(m,n,\tau)} \in G_{i,\tau}^{\text{Available}}, \\ g_{(m',n',\tau-1)} \in G_{i,\tau-1}^{\text{Available}}, \\ g_{(m',n',\tau-1)} \neq g_i^{\text{Entry}} : \\ \exists e_{(m',n',\tau-1)}^{(m'',n'',\tau-2)} \in E_{i,\tau-1}^{\text{Generated}} \end{array} \right\}, \tau \in T_i, \tau \neq \tau_i^{\text{Entry}} \quad (6)$$

where $e_{(m',n',\tau)}^{(m,n,\tau)}$ represents the directed edge from point $c_{(m,n,\tau)}$ to point $c_{(m',n',\tau+1)}$. $E_{i,\tau}^{\text{Generated}}$ represents the set of generated directed edges between the adjacent time step planes ($\tau - 1$) and τ .

For the set of candidate directed edges $E_{i,\tau}^{\text{Candidate}}$, the speed constraint can be represented as:

$$V_i^{\min} \Delta T \leq |c_{(m',n',\tau-1)} - c_{(m,n,\tau)}| \leq V_i^{\max} \Delta T \quad (7)$$

where V_i^{\min} is UAV i 's minimum speed. Obviously, this constraint is compatible with fixed-wing aircraft. If the UAV is a rotorcraft and is allowed to hover, V_i^{\min} will be set to 0.

The reachability constraint can be represented as:

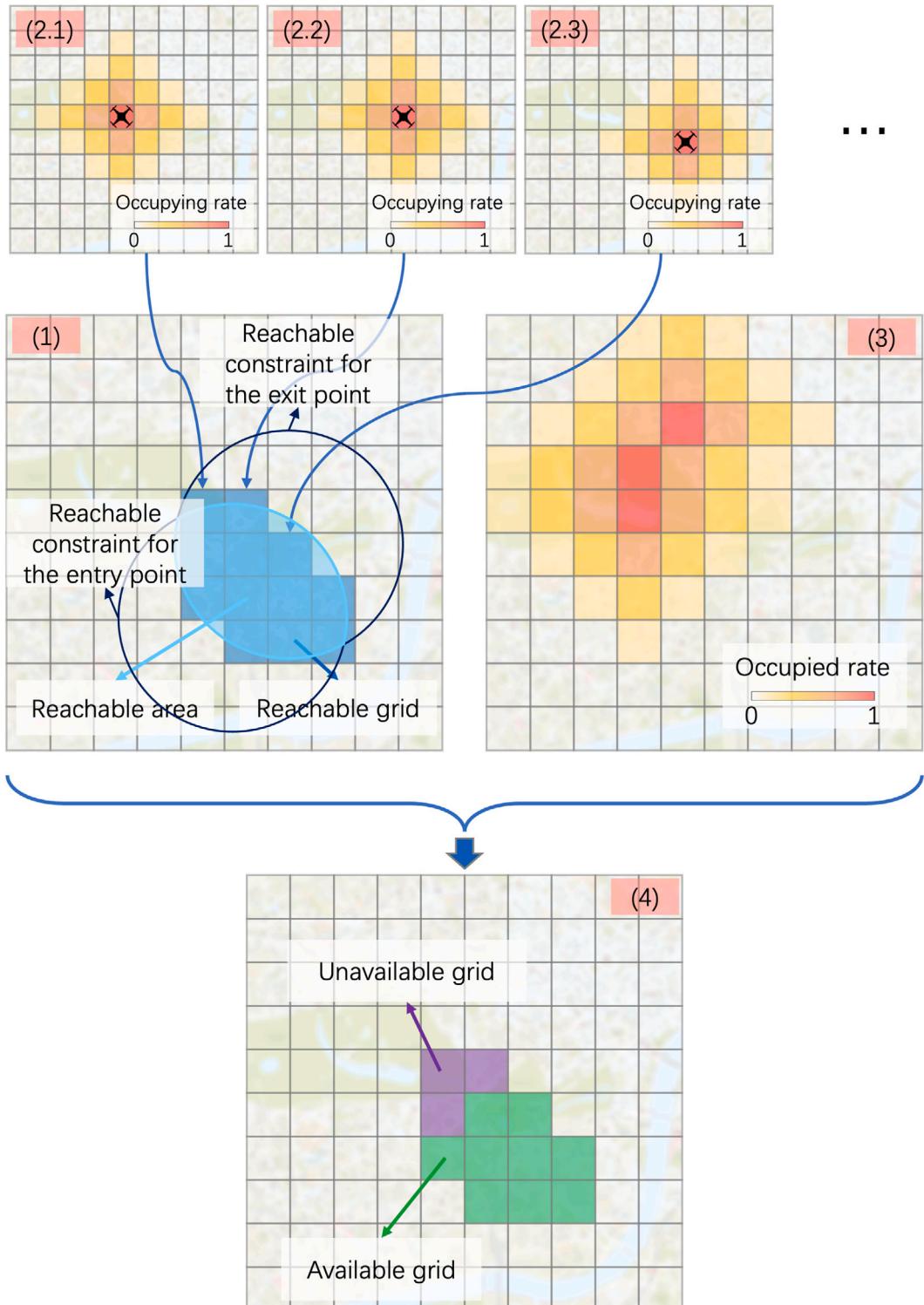
$$\min \{ |m' - m|, |n' - n| \}, (m', n') \in \{g_i^{\text{Exit}}\} \cup G_i^{\text{AE}} \leq \tau_i^{\text{Exit}} - \tau \quad (8)$$

It should be noted that the reachability constraint is not necessary for generating a directed graph that can be used to search for conflict-free trajectory, but it can effectively reduce the size of the graph and thus improve the path search speed.

Since this paper assumes that the airspace unit is provided with several fixed entry and exit points, we define the grid at which the entry point is located as the first entrance protection grid, the grid outside it as the second entrance protection grid, and so on (refer to Fig. 9 as an example). In practice, several layers of entrance protection grids can be set according to the grid size and actual needs. We require the UAV to move outward from the entry grid at least one layer of grids per time step after entering the airspace unit until it leaves the area of entrance protection grids. Therefore, the entrance constraint can be presented as:

$$(m', n', \tau) \notin G_{k,\tau}^{\text{Protection}}, \tau \geq \tau_i^{\text{Entry}} + k, k \in K \quad (9)$$

where $G_{k,\tau}^{\text{Protection}}$ is the set of k th entrance protection grids at time step τ , and K is the set of layers of the entrance protection grids (i.e., $K = \{1, 2, \dots, k_{\max}\}$ and k_{\max} is the maximum number of layers of the entrance protection grids). It should be noted that

**Fig. 7.** Method of obtaining available grids.

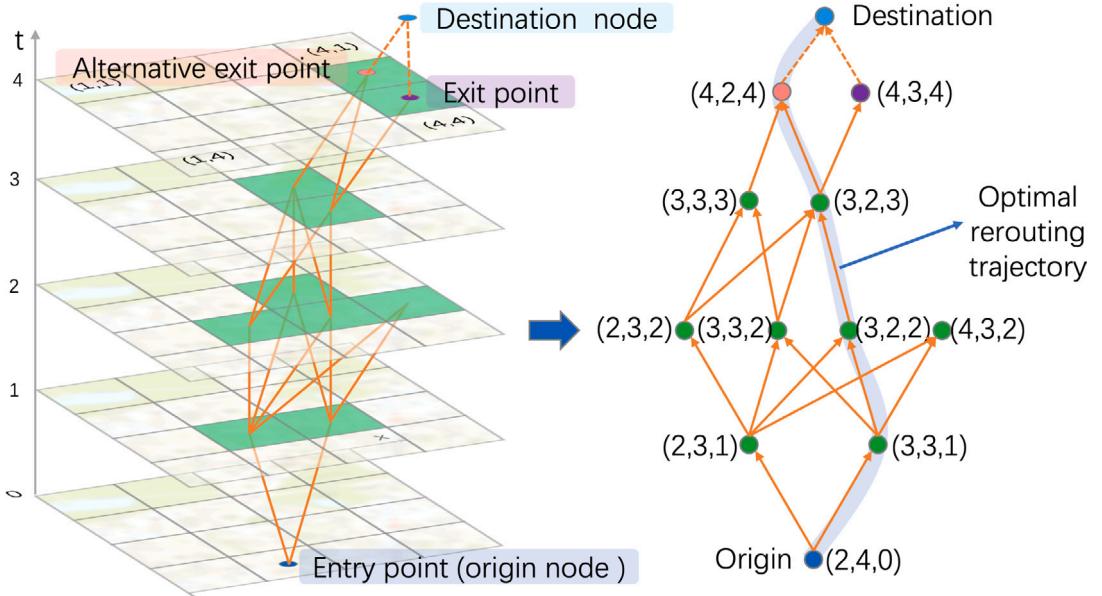


Fig. 8. Generation method of spatio-temporal reachable graph.

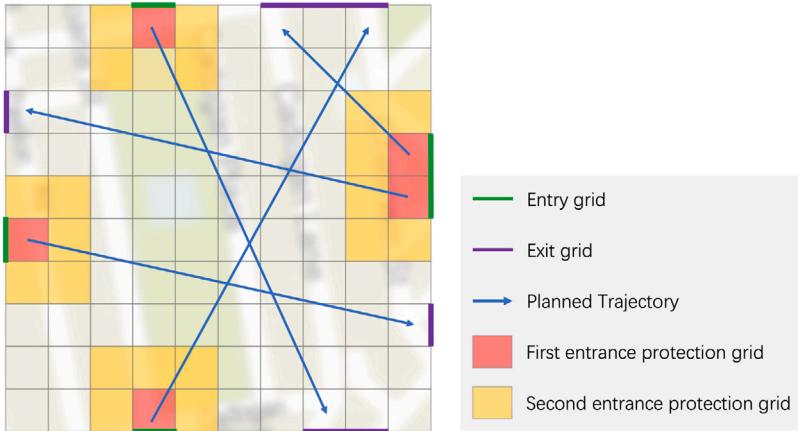


Fig. 9. Entrance protection grids.

in order to simplify the problem, this paper uses a regularised way to build the entrance constraint. In practice, if the CISPs can synchronise specific information about subsequent UAVs entering the airspace unit within the flight time step of the current UAV to the ATMSPs serving it, more accurate entrance constraints can be constructed to improve the efficiency of airspace utilisation.

Since this paper assumes that the change of UAV motion state is instantaneous, in order to make the model closer to the real situation, it is better to consider the restriction of UAV yaw angle, especially when the grid size is small. This paper requires the UAV to move only in a direction close to the exit point, which is also conducive to flight efficiency. Therefore, the direction constraint can be represented as:

$$(m' - m) \left(m_i^{\text{Exit}} - m_i^{\text{Entry}} \right) \geq 0 \quad (10)$$

$$(n' - n) \left(n_i^{\text{Exit}} - n_i^{\text{Entry}} \right) \geq 0 \quad (11)$$

where $(m_i^{\text{Entry}}, n_i^{\text{Entry}})$, $(m_i^{\text{Exit}}, n_i^{\text{Exit}})$ are the grids that UAV i 's entry and exit points locate, respectively. It should be noted that the direction constraint may not be necessary in practice, depending on the UAV's performance and the grid's size. If the performance of the UAV allows it to complete yaw manoeuvres in any direction in one time step, it can be considered that direction constraints are not necessary.

Based on the above connection, speed, reachability, entrance and direction constraints, the set of generated directed edges between the adjacent time step planes ($\tau - 1$) and τ can be represented as:

$$\mathbf{E}_{i,\tau}^{\text{Generated}} = \left\{ e_{(m,n,\tau)}^{(m',n',\tau-1)} \middle| \begin{array}{l} e_{(m,n,\tau)}^{(m',n',\tau-1)} \in \mathbf{E}_{i,\tau}^{\text{Candidate}} : \\ V_i^{\min} \Delta T \leq \overline{|c_{(m',n',\tau-1)} c_{(m,n,\tau)}|} \leq V_i^{\max} \Delta T, \\ \min \left\{ \begin{array}{l} \min \{ |m' - m|, |n' - n| \}, \\ (m', n') \in \{g_i^{\text{Exit}}\} \cup \mathbf{G}_i^{\text{AE}} \end{array} \right\} \leq \tau_i^{\text{Exit}} - \tau, \\ (m', n', \tau) \notin \mathbf{G}_{k,\tau}^{\text{Protection}}, k \in K, k \leq \tau - \tau_i^{\text{Entry}}, \\ (m' - m) (m_i^{\text{Exit}} - m_i^{\text{Entry}}) \geq 0, \\ (n' - n) (n_i^{\text{Exit}} - n_i^{\text{Entry}}) \geq 0 \end{array} \right\}, \tau \in T_i, \tau \neq \tau_i^{\text{Entry}} \quad (12)$$

Therefore, the generated directed graph $\Omega_i^{\text{Digraph}}$ for UAV i is represented as:

$$\Omega_i^{\text{Digraph}} = \left\{ e_{(m',n',\tau+1)}^{(m,n,\tau)} \middle| e_{(m',n',\tau+1)}^{(m,n,\tau)} \in \left(\bigcup_{\tau \in T_i, \tau \neq \tau_i^{\text{Exit}}} \mathbf{E}_{i,\tau}^{\text{Generated}} \right) \cup \mathbf{E}_i^{\text{AE}} \right\} \quad (13)$$

where \mathbf{E}_i^{AE} is the set of edges that connects the destination node and available exit grids, and it can be represented as:

$$\mathbf{E}_i^{\text{AE}} = \left\{ e_{\text{Destination}}^{(m,n,\tau)} \middle| g_{(m,n,\tau)} \in \{g_i^{\text{Exit}}\} \cup \mathbf{G}_i^{\text{AE}} : \exists e_{(m,n,\tau_i^{\text{Exit}})}^{(m',n',\tau-1)} \in \mathbf{E}_{i,\tau_i^{\text{Exit}}}^{\text{Generated}} \right\} \quad (14)$$

It should be noted that the destination node has no physical meaning and is used to implement the function of flexible exit points.

As long as the directed graph $\Omega_i^{\text{Digraph}}$ can be obtained, there must be feasible reroute trajectories. On the contrary, if a set of candidate directed edges $\mathbf{E}_{i,\tau}^{\text{Candidate}}$ is empty (i.e., directed graph $\Omega_i^{\text{Digraph}}$ cannot be generated), there is no feasible reroute trajectories. When unsolvable, an adaptive CTA postponement strategy is introduced (Chen et al., 2022), with the following iterative process: postpone UAV i 's exit time τ_i^{Exit} by N^{Postpone} time steps until a directed graph $\Omega_i^{\text{Digraph}}$ can be generated.

In this paper, the minimum energy consumption is taken as the optimisation objective, so the energy consumption of the flight represented by an edge is taken as the weight of the edge, and then the weighted directed graph (WDG) is obtained. In the instantiation of this paper, we borrow the energy consumption model and relevant parameters from state-of-the-art research on EHang 184's energy consumption (refer to Appendix F for more details) (Pradeep and Wei, 2019). If we use φ to represent the weighted function of the edge and $c_{e_{(m',n',\tau+1)}^{(m,n,\tau)}}^{\text{Energy}}$ represent the energy consumption of edge $e_{(m',n',\tau+1)}^{(m,n,\tau)}$, UAV i 's generated weighted directed graph Ω_i^{WDG} is represented as:

$$\Omega_i^{\text{WDG}} = \left\{ \left(e_{(m',n',\tau+1)}^{(m,n,\tau)}, \varphi \right) \middle| e_{(m',n',\tau+1)}^{(m,n,\tau)} \in \Omega_i^{\text{Digraph}}, \varphi \left(e_{(m',n',\tau+1)}^{(m,n,\tau)} \right) = c_{e_{(m',n',\tau+1)}^{(m,n,\tau)}}^{\text{Energy}} \right\} \quad (15)$$

Since the destination node has no physical meaning, when $e_{(m',n',\tau+1)}^{(m,n,\tau)} \in \mathbf{E}_i^{\text{AE}}$, $c_{e_{(m',n',\tau+1)}^{(m,n,\tau)}}^{\text{Energy}} = 0$.

It should be noted that when constructing a directed graph, the nodes' candidates are always the grid's centre points, so when the UAV is located in different spatio-temporal grids, its occupying rate distribution for the grid near its locating grid is the same. Therefore, we can construct a central occupying rate distribution nominal map for the UAV in advance and call it directly when needed to speed up the calculation (otherwise, it is very time-consuming to repeat the grid-based integral calculation described in Appendix C).

However, when the UAV is not rerouted (e.g., the UAV flies from the entry point to the exit point in a straight line), the UAV is not necessarily located at the centre point of a grid in each time step plane. In order to avoid repeated integral calculation in this case, we introduce a compact occupying rate distribution nominal map based on calculating the worst-case probability distribution. Readers are referred to Appendix G for more details on central and compact occupying rate distribution nominal maps.

3.3.3. Optimal trajectory search

After obtaining the weighted directed graph, this paper uses the A-star algorithm (Hart et al., 1968) to find the optimal path on the graph (i.e., the conflict-free trajectory) quickly. The evaluation function of the A-star algorithm consists of actual cost and heuristic cost:

$$f_{\Omega_i^{\text{WDG}}}^{\text{Value}}(m, n, \tau) = f_{\Omega_i^{\text{WDG}}}^{\text{Actual}}(m, n, \tau) + f_{\Omega_i^{\text{WDG}}}^{\text{Heuristic}}(m, n, \tau) \quad (16)$$

where $f_{\Omega_i^{\text{WDG}}}^{\text{Value}}(m, n, \tau)$, $f_{\Omega_i^{\text{WDG}}}^{\text{Actual}}(m, n, \tau)$ and $f_{\Omega_i^{\text{WDG}}}^{\text{Heuristic}}(m, n, \tau)$ represent the evaluation cost of node (m, n, τ) , actual cost from the origin to node (m, n, τ) , and heuristic cost from node (m, n, τ) to the destination in UAV i 's current weighted directed graph Ω_i^{WDG} , respectively. Actual cost $f_{\Omega_i^{\text{WDG}}}^{\text{Actual}}(m, n, \tau)$ is the optimal path with the minimum energy consumption from the origin to node (m, n, τ) :

$$f_{\Omega_i^{\text{WDG}}}^{\text{Actual}}(m, n, \tau) = \min \left\{ c^{\text{Energy}} \left[(m_i^{\text{Entry}}, n_i^{\text{Entry}}, \tau_i^{\text{Entry}}), (m, n, \tau) \right] \middle| \Omega_i^{\text{WDG}} \right\} \quad (17)$$

where $c^{\text{Energy}} \left[(m_i^{\text{Entry}}, n_i^{\text{Entry}}, \tau_i^{\text{Entry}}), (m, n, \tau) \right] | \Omega_i^{\text{WDG}}$ represents the energy consumption from node $(m_i^{\text{Entry}}, n_i^{\text{Entry}}, \tau_i^{\text{Entry}})$ to node (m, n, τ) on weighted directed graph Ω_i^{WDG} . The A-star algorithm ensures that the optimal solution is obtained under the condition that the heuristic cost is always less than or equal to the actual cost from node (m, n, τ) to the destination. According to rotorcraft aeromechanics, the required power of rotorcraft changes with the flight speed in a saddle-shaped trend, i.e., from hovering to the maximum flight speed, the required power decreases first and then increases after reaching the minimum value (Johnson, 2013). Because the flight time is constant (i.e., $(\tau_i^{\text{Exit}} - \tau)\Delta T$), the actual energy consumption must be greater than or equal to the energy consumption when the UAV is always flying at minimum power. Therefore, the heuristic cost to satisfy the conditions for obtaining the optimal solution can be written as:

$$f_{\Omega_i^{\text{WDG}}}^{\text{Heuristic}}(m, n, \tau) = P_i^{\text{Min,Require}}(\tau_i^{\text{Exit}} - \tau)\Delta T \quad (18)$$

where $P_i^{\text{Min,Require}}$ is UAV i 's minimum required power. Therefore, Eq. (16) can be rewritten as:

$$f_{\Omega_i^{\text{WDG}}}^{\text{Value}}(m, n, \tau) = \min \left\{ c^{\text{Energy}} \left[(m_i^{\text{Entry}}, n_i^{\text{Entry}}, \tau_i^{\text{Entry}}), (m, n, \tau) \right] | \Omega_i^{\text{WDG}} \right\} + P_i^{\text{Min,Require}}(\tau_i^{\text{Exit}} - \tau)\Delta T \quad (19)$$

Then, the optimal rerouting trajectory $T_i^{\text{Rerouting}}$ is obtained, which is represented as:

$$T_i^{\text{Rerouting}} = \left\{ c_{(m_i^{\text{Entry}}, n_i^{\text{Entry}}, \tau_i^{\text{Entry}})}, \dots, c_{(m, n, \tau)}, \dots, c_{(m_i^{\text{Exit}}, n_i^{\text{Exit}}, \tau_i^{\text{Exit}})} \right\} \quad (20)$$

3.3.4. Algorithm design

The proposed method effectively converts the large-scale UAV CD&R problem within an airspace unit into a conflict-free trajectory planning problem based on sequential decision-making (i.e., applying FCFS rules), thereby reducing the complexity of the problem, lowering solution difficulty, and increasing solution speed. Algorithm 1 shows the running process of the conflict-free trajectory search based on a spatio-temporal reachable graph, where the hash sign (#) indicates a comment. The required inputs are UAV i 's planned trajectory in the airspace unit and the set of airspace unit's remaining occupiable rates. The algorithm consists of three main parts: (a) feasible rerouting point generation, (b) weighted directed graph generation and (c) optimal rerouting trajectory search, corresponding to Sections 3.3.1–3.3.3, respectively.

In feasible rerouting point generation, the algorithm first determines whether each grid in the time-step plane is located in the reachable area of UAV i one by one, and it is appended to the corresponding set of reachable grids if it is. Then, the algorithm determines the reachable grids one by one: if the safety constraint is satisfied when UAV i is located in the centre of the grid, the reachable grid is appended to the corresponding set of available grids (line 5). After completing a time step, the algorithm determines: if there is no available grid in that time step, indicating that there is no conflict-free rerouting trajectory under the current planned trajectory, the algorithm marks the current problem unsolvable and stops determining the subsequent time steps (lines 6–8). After the loop of searching available grids stops, if it is due to the unsolvability, the algorithm will execute the adaptive CTA postponement strategy (i.e., the CTA of UAV i 's exit point is postponed by N^{Postpone} time steps) and try to solve the new problem (lines 10–13); otherwise, for the sake of modelling and coding, the algorithm takes UAV i 's entry and exit points as the only grid of the corresponding set of available grids, respectively (line 14).

In the weighted directed graph generation, the algorithm first determines each time step plane in turn according to the time step order: for each available grid in the previous time step plane, if its centre is UAV i 's entry point or the end of an edge in set $E_{i,\tau-1}^{\text{Generated}}$, it is taken as the start of a new edge while the point in the current time step that satisfies the speed, reachability, entrance and direction constraints is taken as its end, and the newly generated edge $e_{(m,n,\tau)}^{(m',n',\tau-1)}$ is appended to set $E_{i,\tau}^{\text{Generated}}$. It should be noted that the algorithm only determines available grids within a specific range (based on UAV i 's speed performance limitation) rather than all available grids to reduce the search space (line 18). Next, if there is no edge generated at time step plane τ , the algorithm marks the current problem unsolvable and stops determining the subsequent time steps; otherwise, the newly generated edges are used to construct directed graph $\Omega_i^{\text{Digraph}}$ (lines 19–24). After the loop of edge generation stops, if it is due to the unsolvability, the algorithm will execute the adaptive CTA postponement strategy and try to solve the new problem (lines 25–28); otherwise, the directed graph $\Omega_i^{\text{Digraph}}$ is generated, and then give weight to its edges to obtain the weighted directed graph Ω_i^{WDG} (line 30).

In the optimal rerouting trajectory search, the algorithm uses the A-star algorithm with Eq. (19) as the evaluation function to search the optimal path of weighted directed graph Ω_i^{WDG} (line 32). Finally, UAV i 's optimal rerouting trajectory $T_i^{\text{Rerouting}}$ is obtained.

Algorithm 1 Conflict-free trajectory search based on spatio-temporal reachable graph

Input: UAV i 's planned trajectory in the airspace unit, i.e., $(m_i^{\text{Entry}}, n_i^{\text{Entry}}, \tau_i^{\text{Entry}})$ and $(m_i^{\text{Exit}}, n_i^{\text{Exit}}, \tau_i^{\text{Exit}})$, the set of airspace unit's remaining occupiable rates $R = \left\{ r^{(m,n,\tau)} \mid g_{(m,n,\tau)} \in G_\tau, \tau \in T_i \right\}$

- 1: Initialise directed graph $\Omega_i^{\text{Digraph}} \leftarrow \emptyset$
- 2: **while** $\Omega_i^{\text{Digraph}} = \emptyset$ **do**
- 3: # Feasible rerouting point generation
- 4: **for** τ from $(\tau_i^{\text{Entry}} + 1)$ to $(\tau_i^{\text{Exit}} - 1)$ **do**
- 5: Generate the set of reachable grids $G_{i,\tau}^{\text{Reachable}}$, and on this basis generate the set of available grids $G_{i,\tau}^{\text{Available}}$

```

6:   if  $G_{i,\tau}^{\text{Available}} = \emptyset$  then
7:     Mark as unsolvable and break the loop
8:   end if
9: end for
10: if unsolvable then
11:    $\tau_i^{\text{Exit}} \leftarrow \tau_i^{\text{Entry}} + N^{\text{Postpone}}\tau$ 
12:   Go to the next loop
13: end if
14:  $G_{i,\tau}^{\text{Available}} \leftarrow \{g_i^{\text{Entry}}\}, G_{i,\tau}^{\text{Exit}} \leftarrow \{g_i^{\text{Exit}}\}$ 
15: # weighted directed graph generation
16: Initialise directed graph  $\Omega_i^{\text{Digraph}} \leftarrow \emptyset$ 
17: for  $\tau$  from  $(\tau_i^{\text{Entry}} + 1)$  to  $\tau_i^{\text{Exit}}$  do
18:   generate the set of generated directed edges  $E_{i,\tau}^{\text{Generated}}$ 
19:   if  $E_{i,\tau}^{\text{Generated}} = \emptyset$  then
20:     Mark as unsolvable and break the loop
21:   else
22:     Append  $E_{i,\tau}^{\text{Generated}}$  to  $\Omega_i^{\text{Digraph}}$ 
23:   end if
24: end for
25: if unsolvable then
26:    $\tau_i^{\text{Exit}} \leftarrow \tau_i^{\text{Exit}} + N^{\text{Postpone}}\tau$ 
27:    $\Omega_i^{\text{Digraph}} \leftarrow \emptyset$ 
28: end if
29: end while
30: Give weight to edge  $e_{(m',n',\tau+1)}^{(m,n,\tau)} \in \Omega_i^{\text{Digraph}}$  by weighted function  $\varphi$ , and then obtain weighted directed graph  $\Omega_i^{\text{WDG}}$ 
31: # Optimal rerouting trajectory search
32: Use the A-star algorithm with (19) as the evaluation function to search the optimal path of weighted directed graph  $\Omega_i^{\text{WDG}}$ .
Output: UAV  $i$ 's optimal rerouting trajectory  $T_i^{\text{Rerouting}}$ 

```

4. Simulation experiments

In this section, we first introduce how we generate experimental scenarios and set relevant parameters (Section 4.1). We then construct six comparative models (Section 4.2) to investigate the differences between the proposed model and conventional similar models while also validating the effectiveness of the key features of the proposed model. Finally, we conduct large-scale random testing on these six models within the generated experimental scenarios. We showcase and analyse performance indicators related to effectiveness, efficiency, stability, and timeliness (Section 4.3). This experiment is conducted using Python 3.9 on a macOS system with an Apple M1 Pro chip and 16 GB of memory.

4.1. Scenario parameter setting

The simulation scenario in this paper is a square airspace unit as shown in Fig. 10, and the scenario parameter setting is shown in Table 2. Fig. 11 illustrates the distribution of occupying rate when a UAV is positioned at the centre of a grid in the experiment, provided for reference. In order to explore the variation rule of the performance of the proposed method with traffic density, six groups of scenarios with different traffic densities (10, 20, 30, 40, 50 and 60 per minute) are randomly generated, with 100 scenarios in each group, totalling 600 scenarios (refer to Appendix H for the planned trajectory generation steps). Fig. 12 shows the initial conflict intensity in the generated scenarios, where the colour shadow represents the error of the corresponding data (standard deviation of the 100 scenario results), and the subsequent performance result plots are also presented in this way. The initial conflict rate is the ratio of the number of initial conflicts to the total number of UAVs in the scenario. It should be noted that since the FCFS rule and the probability-based grid occupancy model are adopted in this paper, for a UAV, if the remaining occupiable rate of a space-time grid in the airspace unit is less than the UAV's required occupying rate, it is counted as a conflict.

4.2. Comparison models

In order to explore the effects of various features of the proposed method on performances, the following six comparison models are designed (Table 3 shows their features):

1. **Model P:** This model is the prototype of the proposed method, which has the following five features:

- Grid occupancy - probabilistic: the UAV's occupancy rate to the grid is calculated based on its position distribution probability so that the UAV is allowed to share the spatio-temporal grid within a certain safety threshold (refer to Sections 3.1 and 3.2).

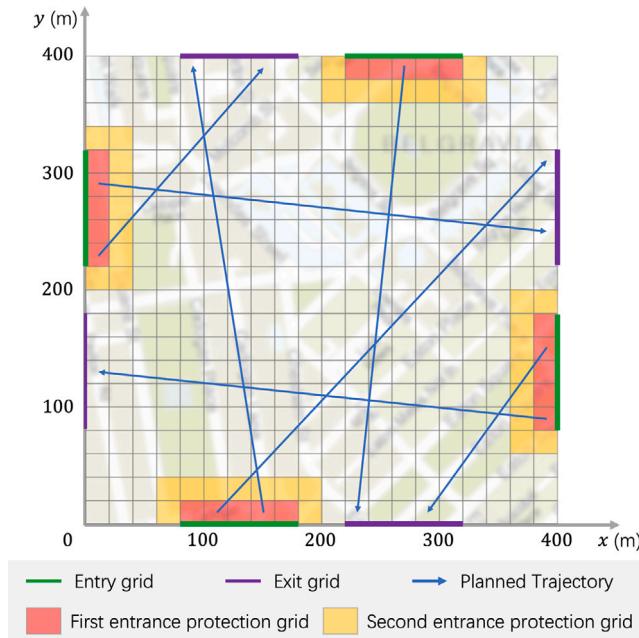


Fig. 10. Experimental scenario diagram.

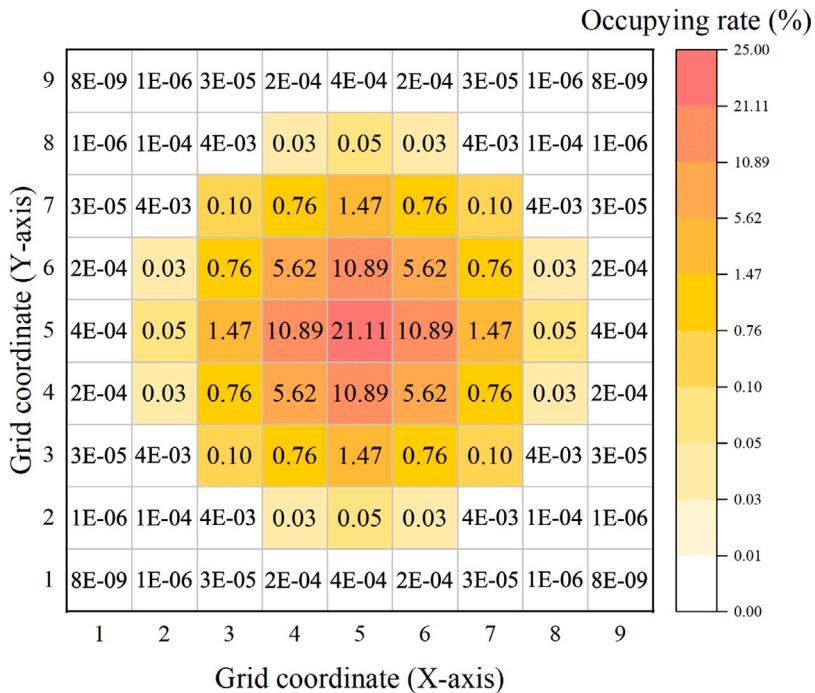


Fig. 11. Instantiation of grid-based UAV position distribution probability. Since occupying rate identification threshold ϕ is set to 0.001, the occupying rate that is less than 0.001 (white block in the heat map) is calculated as zero. Nevertheless, this figure still shows these neglected values in scientific notation for reference.

- Number of rerouting points - multiple: the proposed conflict-free trajectory search based on spatio-temporal reachable graph is equivalent to choosing a maximum of $(\tau_i^{\text{Exit}} - \tau_i^{\text{Entry}} - 1)$ rerouting points (refer to Section 3.3).
- Adaptive CTA postponement strategy: the UAV is allowed to postpone its exit point's CTA in the airspace unit when there is no feasible solution under the current CTA constraint (refer to Section 3.3.2).

Table 2
Scenario parameter setting.

Parameter	Values
Airspace unit size (side length)	400 m
Grid size (side length)	20 m
Number of layers of entrance protection grid	3
Speed range (planned) [$V_{\text{Planned}}^{\min}, V_{\text{Planned}}^{\max}$]	[15, 20] m/s
Speed range (performance) [$V_{\text{Performance}}^{\min}, V_{\text{Performance}}^{\max}$]	[0, 27.8] m/s ^a
Simulation duration	10 min
Time step length ΔT	2 s
Positioning error radius (95% confidence interval) ^b	40 m
Safety threshold P_{Safety} ^c	0.0230
Occupying rate identification threshold ϕ	0.0001
Maximum number of CTA postponement	5

^a Refer to EHang 184 official performance specifications.

^b As can be calculated from Eq. (B.4) (i.e., $D = 40, P = 95\%$), the standard deviation σ is 16.340.

^c The details about how to set the safety threshold are referred to Appendix I.

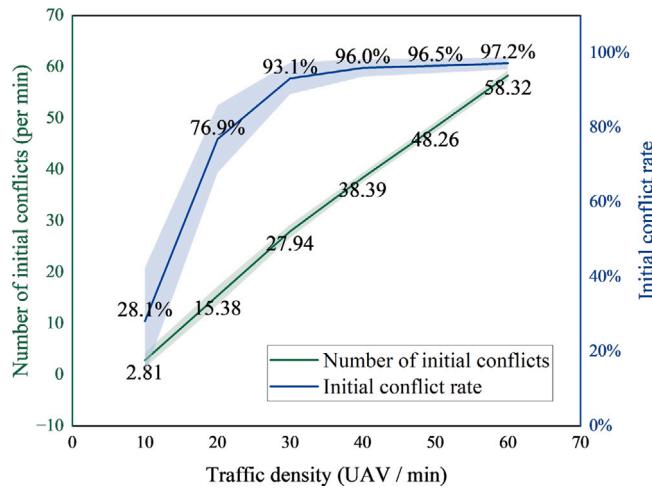


Fig. 12. Initial conflict intensity in the generated scenarios.

- Entrance protection grid: the entrance of the airspace unit is surrounded by several layers of protection grids to prevent UAVs from blocking the entrance (refer to the entrance constraint in Section 3.3.2).
 - Flexible exit grid: when searching for a conflict-free rerouting trajectory, the UAV allows the replacement of the preset exit grid with alternative exit grids to expand the solution space (refer to assumption item 5 in Section 2, and Section 3.3.2).
2. **Model E:** This model uses an entire occupancy model of the grid (i.e., the conventional grid occupancy model), which assumes that the grid within the 95% confidence interval of the positioning error is 100% occupied. It is used to verify the validity of the proposed probability-based grid occupancy model.
 3. **Model S:** This model selects only one grid-based rerouting point to generate a conflict-free trajectory (e.g., our previous research on trajectory planning Chen et al., 2022) and is used to verify the validity of the proposed conflict-free trajectory search based on the spatio-temporal reachable graph.
 4. **Model NAP:** This model does not allow CTA to be postponed to verify the validity of the adaptive CTA postponement strategy.
 5. **Model NEP:** This model does not set entrance protection grids to verify the validity of the entrance protection grid function.
 6. **Model NFE:** This model does not set alternative exit grids to verify the validity of the flexible exit grid function.

It should be noted that the existing UAV conflict-free trajectory planning methods are designed for specific operational environments. Currently, there are still some different opinions regarding UTM operational rules on a global scale, leading to significant differences among various operational environments. Additionally, there are few reports on UAV in-flight fast conflict-free trajectory re-planning methods, such as the proposed method that employs a probability-based spatio-temporal grid occupancy model. Therefore, this study only compares and analyses the proposed method and its variants.

Table 3
Features of comparison models.

Models	Grid occupancy	Number of rerouting points	Adaptive CTA postponement strategy	Entrance protection grid	Flexible exit grid
P	Probabilistic	Multiple	✓	✓	✓
E	Entire	Multiple	✓	✓	✓
S	Probabilistic	Single	✓	✓	✓
NAP	Probabilistic	Multiple	○	✓	✓
NEP	Probabilistic	Multiple	✓	○	✓
NFE	Probabilistic	Multiple	✓	✓	○

Check mark (✓): featured, circle mark (○): not featured.

The names of these models indicate their characteristics, P for probabilistic (i.e., partial occupancy), E for entire (i.e., full occupancy), S for single, N for none, AP for adaptive postponement, EP for entrance protection, and FE for flexible exit.

Table 4
Description of performance test indicators.

Performances	Indicators	Formulas
Effectiveness	Number of unsolved conflicts	n^{Uns}
	Conflict resolution success rate	$(n^{\text{UAV}} - n^{\text{Uns}})/n^{\text{UAV}}$
Efficiency	Number of adjusted trajectories	n^{Adj}
	Average extra energy consumption	$\bar{e}_{\text{ExtEne}}^{\text{Tot}}/n^{\text{Adj}}$
	Average extra energy consumption rate	$\bar{e}_{\text{ExtEne}}^{\text{Tot}}/\bar{t}_{\text{ExtEne}}^{\text{Tot}}$
Delay-related	Number of delayed UAVs	n^{Del}
	Average delay time	$\bar{t}_{\text{Del}}^{\text{Tot}}/n^{\text{Del}}$
	Average delay time rate	$\bar{t}_{\text{Del}}^{\text{Tot}}/\bar{t}_{\text{IniflTim}}^{\text{Tot}}$
Hover-related	Number of hovering UAVs	n^{Hov}
	Average hovering time	$\bar{t}_{\text{Hov}}^{\text{Tot}}/n^{\text{Hov}}$
Stability	Number of actual conflicts	n^{ActCon}
	Domino effect parameter (DEP)	$(n^{\text{ActCon}} - n^{\text{InicCon}})/n^{\text{InicCon}}$
Timeliness	Average computing time	$\bar{t}_{\text{Com}}^{\text{Tot}}/n^{\text{UAV}}$

4.3. Performance test

In order to fully explore the performance of the proposed method, this paper constructs four performance dimensions as follows:

1. Effectiveness: it evaluates whether the proposed method is effective in resolving potential conflicts.
2. Efficiency: it evaluates the cost of the manoeuvres in terms of energy consumption, delay and hover.
3. Stability: it evaluates the degree of secondary conflict caused by trajectory adjustments (i.e., the downstream impact of rerouting Ding et al., 2018).
4. Timeliness: it evaluates whether computing time can meet tactical requirements.

Under the four performance dimensions, a total of 13 indicators are presented as shown in Table 4 for additional description on the related symbols), where some necessary explanations are given below:

- An unresolved conflict is counted when a UAV recognises a potential conflict but does not search for a viable conflict-free path (using the adaptive CTA postponement strategy if permitted).
- The occupying rate of the grid by the UAV whose conflict is unresolved is calculated on the basis of its planned trajectory, but the trajectory is not included in the calculation of efficiency performance indicators.
- Trajectory adjustment is that the UAV adopts a different trajectory than its planned trajectory by rerouting and/or postponing the CTA of its exit point.
- The domino effect parameter (DEP) calculation method refers to previous studies (Chen et al., 2022). The smaller the DEP, the better the stability.
- The computing time consists of three parts: (a) the time to determine whether there is a conflict for the UAV, (b) the time to generate the conflict-free trajectory (if needed), and (c) the time to update the grid occupied rate of the airspace unit.

4.3.1. Effectiveness

Fig. 13 shows the performance test's statistical results of the effectiveness indicators. Since the data range of model E is much larger than the other models, we focus on the other models in the main figure and use the embedded figure to show the data of model E.

Each model's number of unsolved conflicts increases with the traffic density while the conflict resolution success rate correspondingly decreases. Model P shows the best effectiveness. For model P, when the traffic density is 40 UAV/min or less, all potential conflicts are resolved, and even when the traffic density rises to 60 UAV/min, the conflict resolution success rate is as high as

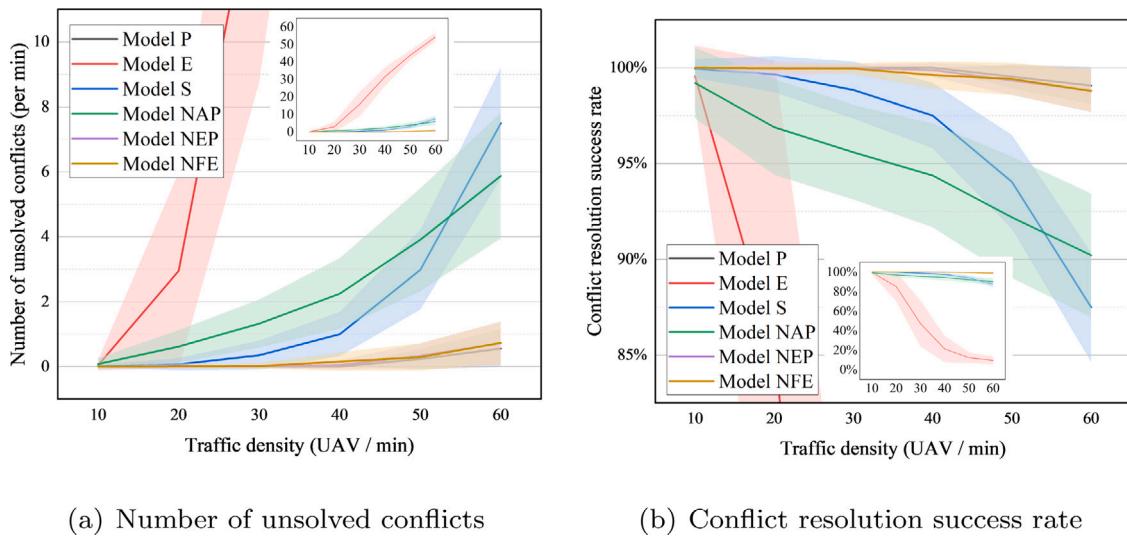


Fig. 13. Performance test results (effectiveness).

99.07%. The effectiveness of models NEP and NFE is similar, slightly worse than that of model P. Due to the lack of functionality, although the conflict resolution success rate of models NEP and NFE can always remain above 98.80% and 98.78%, respectively, they resolve all conflicts only when the conflict density is 10 UAV/min. The effectiveness of models S and NAP is significantly worse than that of models P, NEP and NFE. The conflict resolution success rate of model S decreases exponentially with traffic density, while that of model NAP decreases linearly. When the traffic density is 10 UAV/min, the success rate of model S is 99.95%, while that of model NAP is only 99.2%, but when the traffic density is 60 UAV/min, the success rate of model S (87.50%) is lower than that of model NAP (90.21%).

The results of model S show that the larger solution space caused by the multi-reroute-point feature effectively promotes conflict resolution, and this benefit becomes increasingly obvious with traffic density. The results of model NAP show that the adaptive CTA postponement strategy is an important way to generate conflict-free trajectories, which has an important impact when the traffic density is still low. Although model E's conflict resolution success rate is 99.55% when the traffic density is 10 UAV/min, it decreases sharply with the traffic density, and its success rate is only 9.49% when the traffic density is 60 UAV/min. This is because Model E uses a more conservative conflict detection criterion, so the UAV is probably recognised as an inevitable conflict at its entry grid when the traffic density increases to a certain point while its entry grid is unchangeable in this paper. This also shows that probability-based models have the potential to be applied to scenarios with higher traffic density.

In summary, model P shows the most competitive potential in terms of effectiveness, where the proposed conflict-free trajectory search based on a spatio-temporal reachable graph and the adaptive CTA postponement strategy make major contributions, entrance protection grids and flexible exit grids make minor contributions, and the probability-based grid occupancy model significantly increases its application potential in high-density scenarios.

It should be noted that since the conflict resolution effectiveness of model E is extremely low when the traffic density is greater than 20 UAV/min, only the data of model E when the traffic density is 20 UAV/min or below is usually displayed in the subsequent performance test plots to reduce the data of no reference significance.

4.3.2. Efficiency

Fig. 14 shows the performance test's statistical results of the energy-consumption-related efficiency indicators. Since any adjustment of the trajectory may change the UAV's energy consumption, the number of adjusted trajectories is presented.

The number of adjusted trajectories for each model increases with traffic density (refer to Fig. 14(a)), apparently due to the need to resolve the increasing potential conflicts. The order of the number of adjusted trajectories of each model roughly is as follows: model NAP < model P ≈ model NEP < model NFE < model S, while model E is ignored due to its dramatically low conflict resolution success rate. The model NAP has the smallest number of adjusted trajectories because its conflict resolution success rate is low, i.e., the number of unresolved trajectories n^{Uns} is large, and $n^{Adj} = n^{ActCon} - n^{Uns}$. Although model S's conflict resolution success rate is also low, it has the largest number of adjusted flight trajectories because the conflict-free flight trajectories generated by it cause more secondary conflicts (refer to Section 4.3.3), i.e., the number of actual conflicts n^{ActCon} is larger.

The average extra energy consumption and the average extra energy consumption rate of each model show a similar change trend with the traffic density (refer to Figs. 14(b) and 14(c)). Models P, NAP, NEP and NFE remain stable or increase slightly with traffic density, while model S increase significantly. This is because the single-reroute-point trajectory is not as flexible as the multi-reroute-point trajectory, where the single-reroute-point trajectory often resolves the conflict by flying around a large area. Models E and NFE are more energy-consuming than models P, NAP and NEP because Models E and NFE have smaller solution

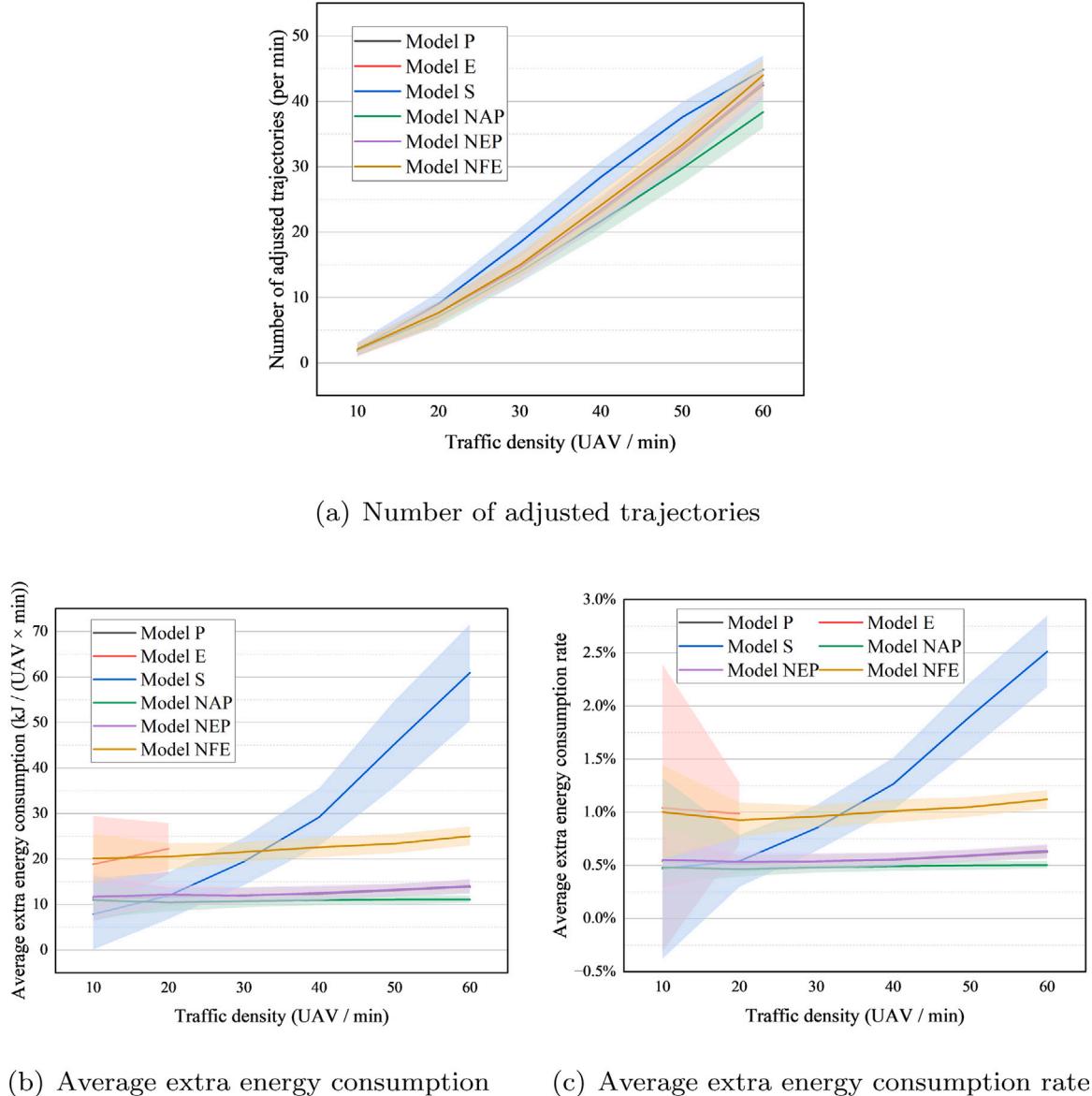


Fig. 14. Performance test results (efficiency: energy consumption).

space, resulting in long rerouting trajectories. Model NAP has the lowest average extra energy consumption (rate) since it does not allow the exit point's CTA postponement, i.e., longer flight time, which usually means more energy consumption. The average extra energy consumption rate for all models except Model S does not exceed 1.2%, which we believe is generally acceptable in actual operation.

Fig. 15 shows the performance test's statistical results of the delay-related efficiency indicators. Since model NAP does not adopt the adaptive CTA postponement strategy, it shows zero in each delay-related indicator and is not discussed here.

The number of delayed UAVs in each model (excluding model NAP) increases with the traffic density because as the conflict intensity increases, more solution space needs to be obtained through the adaptive CTA postponement strategy (refer to Fig. 15(a)). To some extent, the number of delayed UAVs reflects the size of the solution space of the model because the model with a larger solution space is usually more likely to find a feasible solution without delaying the exit point's CTA, which is an operation to expand the solution space. Since the entrance protection grid has little effect on the solution space size, model NEP has a similar number of delayed UAVs to model P, i.e., they have the maximum solution. When the traffic density is less than 30 UAV/min, the number of delayed UAVs of model S is less than that of model NFE, but then, with the increase of traffic density, the number of delayed UAVs of model S is gradually greater than that of model NFE. This may indicate that, in low-density scenarios, flexible exit points are more effective for conflict resolution, while in high-density scenarios, multiple rerouting points are more important.

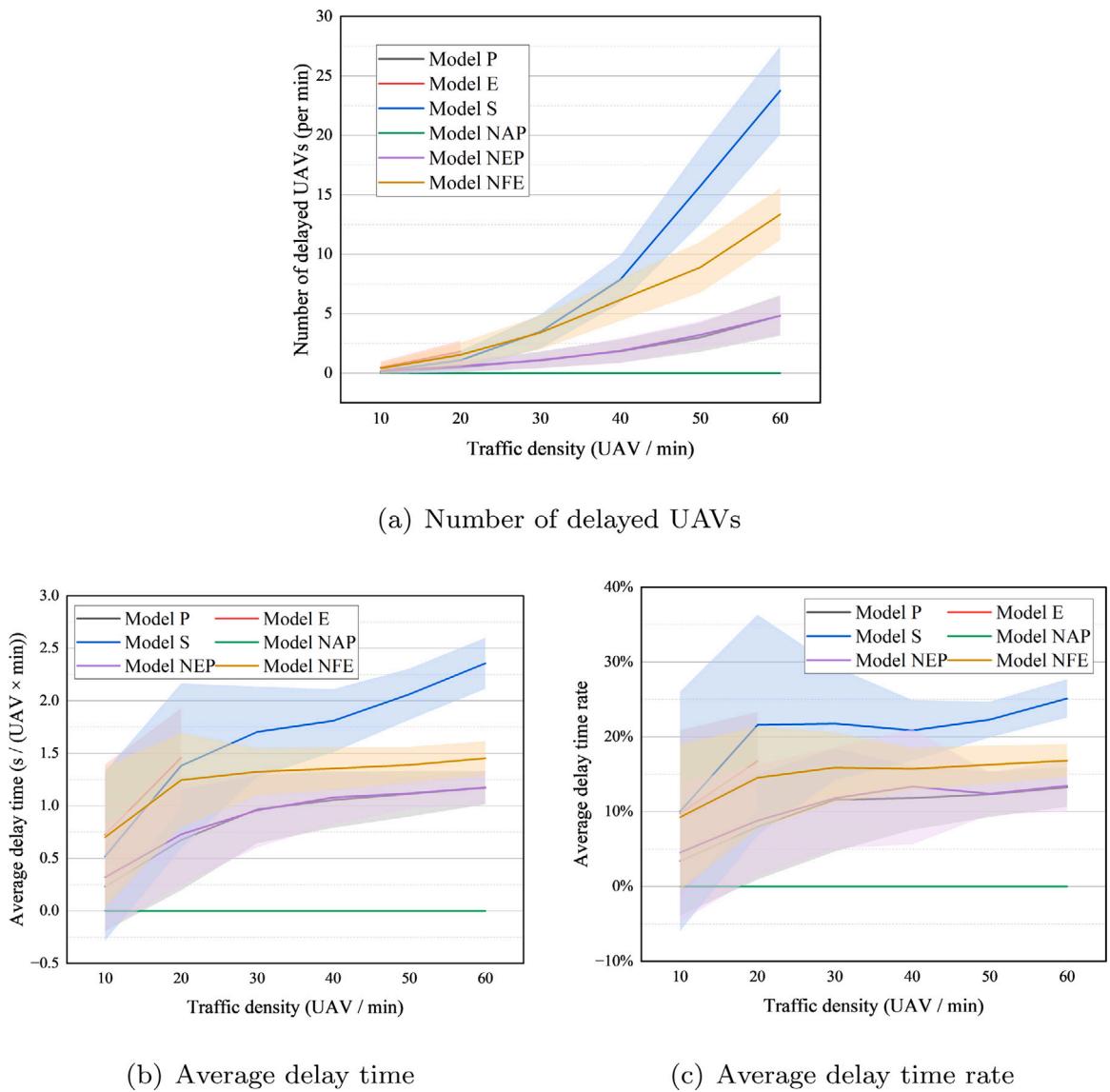


Fig. 15. Performance test results (efficiency: delay).

Model E has the largest number of delayed UAVs when the traffic density is 10 and 20 UAV/min because its solution space is the smallest.

The average delay time and the average delay time rate of each model show a similar change trend with the traffic density (refer to Figs. 15(b) and 15(c)). The order of the delay times for each model is similar to the order of the number of delayed UAVs corresponding to them. Model P has the minimum average delay time (rate), and model NEP is slightly worse than model P, maybe because, in model NEP, UAV hovering at the exit negatively impacts traffic order. When the traffic density is greater than 20 UAV/min, the average delay time (rate) of model S is greater than that of model NFE, which further demonstrates the importance of multiple rerouting points in high-density scenarios. Taking the scenario with the highest density where model P can 100% resolve the conflicts in the experiment (i.e., 40 UAV/min) as an example, the average delay time rate of model P is 11.82%, and considering that only 1.85 UAVs are delayed on average, the total delay time only accounts for about 0.55% of the total flight time of UAVs, which we believe is acceptable in actual operation.

Fig. 16 shows the performance test's statistical results of the hover efficiency indicators. Since model S does not allow hover, it shows zero in each hover-related indicator. It should be noted that the single rerouting point model can actually support hovering at the entry point (i.e., taking the entry point as the rerouting point), but it finally does not support hovering due to the entrance protection grid.

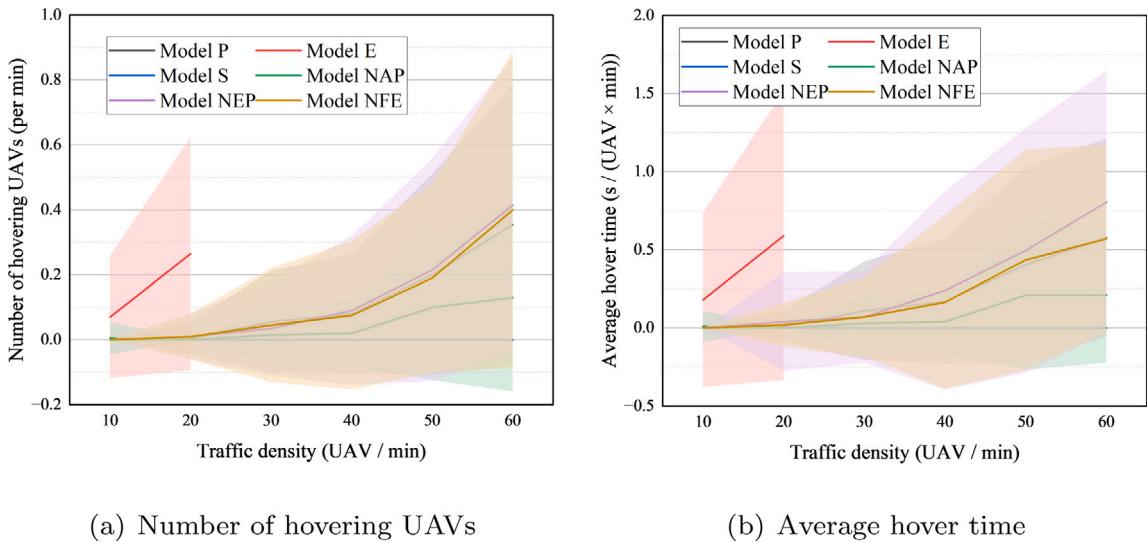


Fig. 16. Performance test results (efficiency: hover).

Table 5
Efficiency indicators of model P when the traffic density is 40 UAV/min.

Indicators	Values
Number of adjusted trajectories	23.03 per min
Average extra energy consumption	12.35 kJ/(UAV × min)
Average extra energy consumption rate	0.55%
Number of delayed UAVs	1.85 per min
Average delay time	1.06 s/(UAV × min)
Average delay time rate	11.82%
Number of hovering UAVs	0.08 per min
Average hover time	0.17 s/(UAV × min)

The number of hovering UAVs and the average hover time of each model show a similar change trend with the traffic density. Except for model S, the model NAP has the smallest number of hovering UAVs and the smallest average hover time because, in the proposed model, hover is likely to require more flight time, and the model NAP does not allow increased flight time by postponing its exit point's CTA. Models P and NFE have similar results on these two hover-related indicators, while model NEP is slightly greater than them (mainly in high-density scenarios). This is because the model NEP does not have the constraint of entrance protection grids, and there is more opportunity to use hover in high-density scenarios, although this has the potential to block the entrance.

In low-density scenarios (such as when the traffic density is less than or equal to 20 UAV/min), model E's number of hovering UAVs and average hovering time are much higher than those of other models, which are 0 or close to 0, because model E's solution space is very small, and hovering has to be adopted to resolve conflicts. For models P, NEP and NFE, even when the traffic density is 60 UAV/min, the number of hovering UAVs is only about 0.4, accounting for only 0.67% of all UAVs, which indicates that hovering is not the main way of conflict resolution in the experimental scenarios. This may be the result of a combination of flight time limitations (CTA constraints) and the optimisation objective (minimum energy consumption), i.e., hovering usually requires extended flight time, and hovering is more energy-consuming than moving in many cases (refer to Fig. F.21 in Appendix F).

Table 5 shows the efficiency indicators of model P when the traffic density is 40 UAV/min (the maximum traffic density at which it can 100% resolve conflicts) for readers' reference. In summary, the following preliminary conclusions regarding effectiveness can be drawn:

1. Compared to the total initial energy consumption and flight time of all UAVs, the proposed method's extra energy consumption and delay time are acceptable.
2. Compared with the extra energy consumption, the delay time is the main cost of trajectory adjustment.
3. Compared with rerouting, hovering is not the main means of conflict resolution.

4.3.3. Stability

Fig. 17 shows the performance test's statistical results of the stability indicators. Since the number of initial conflicts is the same for all models, their orders of the number of actual conflicts and DEP are the same for each traffic density, respectively.

The actual number of conflicts of each model increases with traffic density, while DEP decreases when traffic density is less than 20 UAV/min and then increases. This is because the distribution of entry and exit points in the experiment scenarios is fixed,

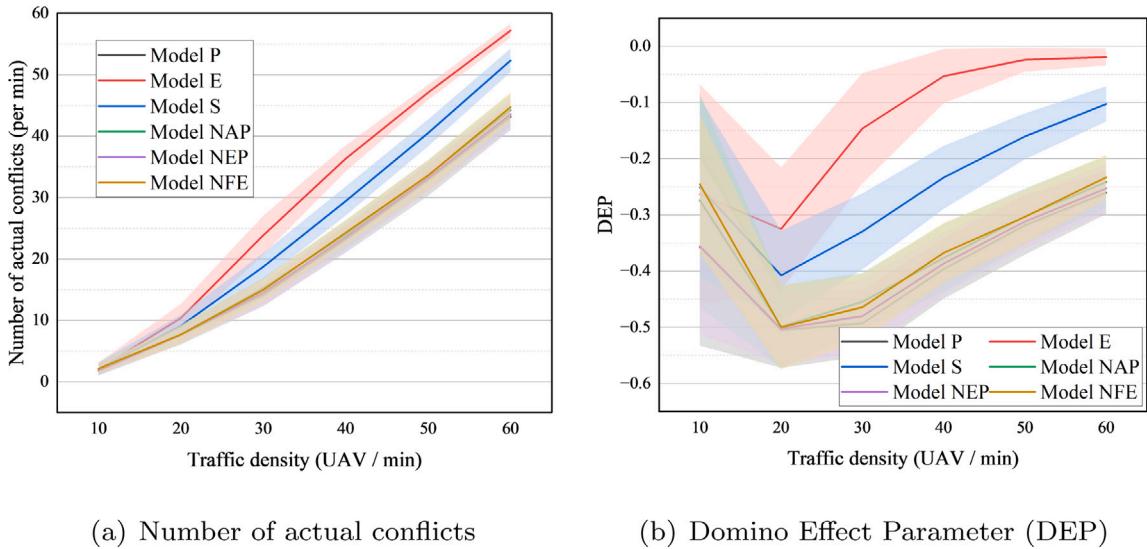


Fig. 17. Performance test results (stability).

resulting in a relatively fixed traffic flow structure and location distribution of the main conflict area. When a UAV makes a trajectory adjustment, it is possible to avoid the main conflict area and resolve multiple conflicts at one time (one or multiple subsequent UAVs change from conflict to conflict-free due to the UAV's trajectory adjustment). Furthermore, when the traffic density is very low, due to the relatively small number of potential conflicts, it is difficult for one trajectory adjustment to resolve multiple conflicts (it is likely to have no conflicts with subsequent UAVs), while when the traffic density is very high, it is difficult to resolve potential conflicts of subsequent UAVs through the current UAV's trajectory adjustment due to a large number of messy conflicts. Therefore, DEP shows this trend of first decreasing and then increasing.

Models P, NAP, NEP and NFE have close results for the two stability indicators, while models E and S are significantly worse than others, where model P is the best and model E is the worst. It is worth noting that although the conflict resolution success rate of model NAP is not ideal and is not as good as model S in most cases (refer to Fig. 13(b)), its DEP is much lower than that of model S when traffic density is greater or equal to 20 UAV/min, indicating that the multi-rerouting-point model makes an important contribution to reducing DEP.

In summary, all the features of the proposed model are helpful in maintaining stability, while the proposed conflict-free trajectory search based on a spatio-temporal reachable graph plays a key role.

4.3.4. Timeliness

Fig. 18 shows the performance test's statistical results of the timeliness indicator. Since the data range of model S is much larger than the other models, we focus on the other models in the main figure and use the embedded figure to show the data of model S.

Model S has a much longer computing time than other models because of their different optimisation search mechanisms. Specifically, the optimisation search process of model S is as follows:

1. Determine whether all the reachable grids on the time-step plane between the entry and exit points are feasible rerouting grids one by one (that is, determine whether a conflict-free trajectory can be generated if the candidate grid is used as a rerouting grid), and calculate its cost.
2. If feasible rerouting grids exist, select the one with the lowest cost as the optimal rerouting grid and generate the conflict-free trajectory accordingly.
3. If no feasible rerouting grid exists, postpone the exit point's CTA and repeat the above steps until a feasible rerouting grid is found or the CTA postponement limit is reached.

In each search, if there are n reachable grids, model S have to carry out the conflict detection for n candidate trajectories one by one, which is very time-consuming. In contrast, the method proposed in this paper avoids conflict detection on a number of candidate trajectories, and the directed graph is generated layer by layer. When a layer fails to generate an edge, it can be directly determined that there is no feasible solution, which also speeds up the calculation. Model E has the fastest computation speed because it does not require integral and grid occupancy rate calculations. In this paper's model, a search of a similar size is repeated for every postponement of the exit point's CTA, and since the model NAP does not support CTA postponement, it is also computationally fast, second only to model E. For the same reason, the computing time order of models P, NEP, and NFE matches their corresponding delay-related indicators (refer to Fig. 15).

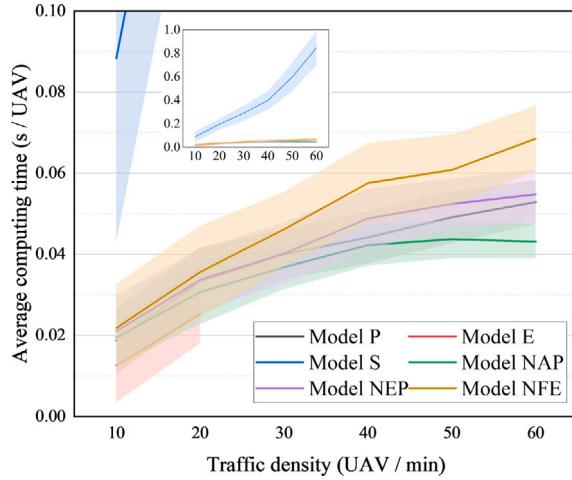


Fig. 18. Performance test results (timeliness).

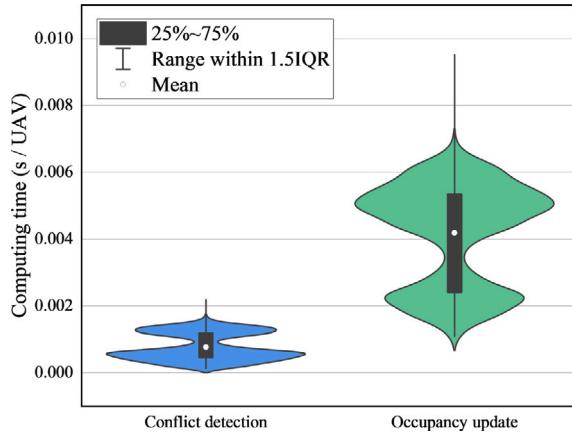


Fig. 19. Computing time of conflict detection and occupancy update.

The average computing time (including conflict detection, solution and occupancy update time) for each UAV is 0.04s for model P when the traffic density is 40 UAV/min (the maximum traffic density at which it can 100% resolve conflicts), which meets the tactical operation requirements, and Fig. 19 shows its computing times of conflict detection and occupancy update. This figure illustrates the distribution of results on these two metrics across 100 scenarios, with wider regions indicating denser distributions. IQR refers to the interquartile range. The average conflict detection and occupancy update times are 0.000766 s and 0.00418 s, respectively. Neither the conflict detection time nor the occupancy update time is normally distributed, but there are two clustered segments. For conflict detection, this is because conflicts are detected one by one according to the time step plane, and the result is returned once the conflict is detected at a time step plane. In this paper's scenarios, the conflict is mostly distributed near the entry or exit point, and the conflict detection computing time is less for the UAV that has detected the conflict near the entrance. For occupancy updates, this is because the UAV uses the central and compact occupying rate distribution nominal maps, respectively, when it is not rerouted and when it is rerouted (refer to Appendix G), and the compact one occupies more grids than the central one (based on the occupying rate identification threshold), thus requiring more occupancy update computing time.

In summary, compared with the single-rerouting-point model, the multi-rerouting-point model (the proposed conflict-free trajectory search based on a spatio-temporal reachable graph) greatly improves the calculation speed and satisfies the timeliness requirement of tactical operation. In addition, the cost of the adaptive CTA postponement strategy is increased computation time to some extent.

5. Discussions

In this study, we learned some interesting and noteworthy lessons. The features of the proposed method are demonstrated effective. Specifically, the probability-based grid occupancy model can be more flexible and reasonable in detecting conflicts and

improving airspace utilisation efficiency compared with the entire grid occupancy model. The multi-rerouting-point model has a larger solution space and a more flexible rerouting trajectory than the single-rerouting-point model. The adaptive CTA postponement strategy is an important means to expand the solution space, but it costs increasing computing time to some extent. Entrance protection grids prevent the UAV from hovering near the entrance and thus help the subsequent UAV entering the airspace unit avoid the inevitable conflict sometimes. Flexible exit grids provide more options at the exit to match low-cost rerouting trajectories. Unlike grid-based methods that can only select four or eight surrounding grids as the next step, this approach can consider all grids in the plane, providing more flexible manoeuvring directions and a larger solution space. Additionally, we find that hovering is not a major means of conflict resolution because hovering may cause more energy consumption and flight time in this paper's scenario, which conflicts with the optimisation objective and the CTA constraint of the constructed optimal problem.

Although this paper uses some instantiation operations to simplify the problem, the proposed method is compatible and scalable. Considering generalisation, although the UAV studied in this paper is instantiated as a quad-rotor aircraft (i.e. EHang184), the proposed method can be easily compatible with fixed-wing aircraft. A compatible method is to set the minimum flight speed to a non-zero value when generating the edges of a directed graph (refer to Eq. (7)). Therefore, it also has potential application in the air traffic management of commercial aviation, but the compatibility and coordination of various parameters need to be further explored. Regarding optimisation, the proposed method can be compatible with customised optimisation objectives, such as flight distance and crash risk (Blom et al., 2021) by adjusting the edge weight of the generated directed graph, and the multi-objective optimisation can also be achieved by weighted summation of them. Since the trajectory planning process of each UAV is independent, the proposed method can also set a differentiated optimisation objective for each UAV. Since the generated directed graph is standard, it can be compatible with any path search algorithms applied to directed graphs, such as various improved A-star algorithms (Erke et al., 2020). Additionally, the proposed method can be compatible with this function by setting the grid occupied by the no-fly zone to the unavailable grid. The modelling of the in-flight trajectory planning problem in this paper is similar to that of pre-flight so that it can be applied to pre-flight trajectory planning problems; still, the uncertainty parameters may need to be adjusted or the uncertainty model reconstructed.

Although this paper makes a meaningful attempt at the method of UAV in-flight trajectory re-planning, as our first step for the study, the proposed method may have limitations or face challenges. For example, since this paper uses a single UAV model (EHang184), a specific airspace structure, grid size and positioning errors, the proposed method in more diverse scenarios is worth exploring further. This paper only considers the positioning error for uncertainties and defines its model as a two-dimensional Gaussian distribution with the same standard deviation in the x and y directions, which makes UAV position distribution non-directional and thus facilitates the distribution model application to the grid model (mainly simplifying the calculation). Additionally, the proposed method is worth further extended to be compatible with directional or irregular probability distribution models so as to deal with multiple uncertainties (e.g., wind and mechanical error) in the real world. For example, based on the worst case, the actual probability distribution model is approximated to a non-directional probability distribution model, i.e., by relaxing the model to ensure that the occupying rate distribution nominal map remains valid, and its performance should be further evaluated to avoid excessive waste of airspace resources. Moreover, the experiments in this paper only verify the feasibility of the proposed method in a single airspace unit. The airspace unit constructed in this paper's experimental scenarios is convenient to form a larger range of airspace, and the entry and exit areas of each adjacent airspace unit are coupled (the entry/exit area of one airspace unit is exactly the exit/entry area of its adjacent airspace unit). Therefore, the coordination of the proposed method in global operation is expected to be further verified in larger airspace (consisting of multiple airspace units). To validate the practical application potential of the proposed method, the flyability of the planned trajectories (flight control feasibility) requires further exploration in a high-fidelity simulation environment or the real world.

6. Conclusions

This paper introduces a UAV in-flight fast conflict-free trajectory re-planning approach considering uncertainty and energy consumption. Through extensive experiments in high-density and complex scenarios, the following key findings have been obtained:

1. The proposed method is effective and efficient in conflict detection and resolution, meeting tactical requirements in terms of computing time. In a $400 \text{ m} \times 400 \text{ m}$ airspace unit with a density of 40 UAVs per minute, this method achieves a 100% success rate in conflict resolution.
2. The effectiveness of various features of the proposed method (i.e., the probability-based grid occupancy model, the multi-rerouting-point model, the adaptive CTA postponement strategy, entrance protection grids and flexible exit grids) has been validated.
3. The method presented in this paper demonstrates superior performance in experiments compared to conventional entire occupancy models of the grid and single-rerouting-point models, thus exhibiting greater potential for practical applications.

In future research, we will further explore the applicability of the proposed methods in broader scenarios, such as diverse UAVs, airspace structures, grid sizes, etc. Expanding the proposed methods to make them applicable to more general uncertainty models is significant for practical application. Additionally, it is necessary to validate the sustainability of the proposed methods over extended periods and larger airspace areas, composed of multiple airspace units. The proposed methods also have the potential to provide theoretical support for the development of pre-flight trajectory planning methods for UAVs. After further consideration of demand and capacity balance constraints, there is potential to develop trajectory plans that can be quickly generated before UAV takeoff, simultaneously satisfying demand and capacity balance constraints, as well as conflict-free constraints. The feasibility of the proposed method will be explored in the real world once UTM operational regulations are sufficiently established and physical operating conditions for large-scale UAV deployment are in place.

CRediT authorship contribution statement

Yutong Chen: Writing – original draft, Validation, Methodology. **Yan Xu:** Writing – review & editing, Supervision, Methodology. **Lei Yang:** Writing – review & editing, Supervision. **Minghua Hu:** Writing – review & editing, Supervision.

Acknowledgements

This research was partially funded by the National Natural Science Foundation of China under grant agreement No. 52402395, the UKRI Future Flight project Stackable Programme on Advanced Air Mobility under grant agreement No. 10069102 and the SESAR JU under grant agreement No. 891965, as part of the European Union's Horizon 2020 research and innovation programme: Artificial Intelligence Solutions to Meteo-Based DCB Imbalances for Network Operations Planning.

Appendix A. Nomenclature

See [Table A.6](#).

Appendix B. Probability distribution model of UAV position

$p_i(t)$ presents the position of UAV i at time t :

$$p_i(t) = (x_i^t, y_i^t) \quad (\text{B.1})$$

where x_i^t and y_i^t represent the horizontal and vertical coordinates, respectively. Since this study only considers positioning error in flight uncertainty, we assume that the position distribution of the UAV follows a two-dimensional Gaussian distribution. Given that the position of the UAV's planned trajectory point at some time is (X, Y) , the probability distribution function of the position $f(x, y)$ is represented as:

$$f(x, y) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} e^{\frac{1}{2(1-\rho^2)}[\frac{(x-X)^2}{\sigma_1^2} - 2\rho\frac{(x-X)(y-Y)}{2\sigma_1\sigma_2} + \frac{(y-Y)^2}{\sigma_2^2}]} \quad (\text{B.2})$$

where σ_1 and σ_2 are the standard deviations for the two dimensions, and ρ is their covariance. This study further assumes that positioning errors are non-directional, so the probability distribution in both dimensions is homogeneous and independent (i.e., $\sigma_1 = \sigma_2$, and $\rho = 0$). Then, Eq. [\(B.2\)](#) can be rewritten as:

$$f(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-X)^2+(y-Y)^2}{2\sigma^2}} \quad (\text{B.3})$$

where $\sigma = \sigma_1 = \sigma_2$. [Fig. 3](#) shows a diagram of the probability distribution of a UAV's position. It should be noted that the probability of the UAV being at any position point on the plane is greater than 0. Still, if the point is farther away from the planned trajectory point of the UAV, the probability is closer to 0.

The value of σ is related to the positioning error. Assume that the positioning error is that there is a probability of P that the actual position of the UAV is in a circular area with the planned trajectory point as the centre and radius D . Therefore, the value of σ can be represented by Eq. [\(B.4\)](#) and be determined according to the desired precision.

$$\sigma = \arg \min_{\sigma} \left| \int_{-D}^{D} \int_{-\sqrt{D^2-y^2}}^{\sqrt{D^2-y^2}} f_{X=0, Y=0}(x, y) dx dy - P \right| \quad (\text{B.4})$$

Appendix C. UAV's occupying model to grid

To balance optimality and computational speed, we discretise the airspace on the horizontal plane based on grids. In order to facilitate the use of the conflict detection model based on grid probability occupancy, this paper takes the integral of the probability of the position distribution of the UAV in a grid as the UAV's occupying rate, as shown in [Fig. 4](#). The grid is a square with sides parallel to the plane coordinate axes x and y , respectively.

The occupying rate of UAV i in grid (m, n) is represented by Eq. [\(C.1\)](#).

$$p_i^{(m,n)} = \int_{Y'_n}^{Y''_n} \int_{X'_m}^{X''_m} f(x, y) dx dy \quad (\text{C.1})$$

m and n are the numbers of the grid in the x and y directions, respectively. X'_m and X''_m represent the x -coordinates of the start and end points of the m th grid, respectively. Y'_n and Y''_n represent the y -coordinates of the start and end points of the n th grid, respectively. In practice, an occupying rate identification threshold ϕ can be set according to actual requirements to facilitate calculation. When the UAV's occupying rate of a certain position is less than the threshold, it takes 0.

Table A.6
Nomenclature.

Symbols	Definitions
$\Omega_i^{\text{Digraph}}$	The generated directed graph for UAV i
Ω_i^{WDG}	UAV i 's generated weighted directed graph
$f_{\Omega_i^{\text{WDG}}}^{\text{Value}}(m, n, \tau)$	The evaluation cost of node (m, n, τ) to the destination in UAV i 's current weighted directed graph Ω_i^{WDG}
$f_{\Omega_i^{\text{WDG}}}^{\text{Actual}}(m, n, \tau)$	The actual cost of node (m, n, τ) to the destination in UAV i 's current weighted directed graph Ω_i^{WDG}
$f_{\Omega_i^{\text{WDG}}}^{\text{Heuristic}}(m, n, \tau)$	The heuristic cost of node (m, n, τ) to the destination in UAV i 's current weighted directed graph Ω_i^{WDG}
$G_{i,\tau}^{\text{Reachable}}$	The set of reachable grids of UAV i at time step τ
G_τ	The set of grids at time step plane τ in the airspace unit
G_i^{AE}	The set of UAV i 's alternative exit grids
$G_{k,\tau}^{\text{Protection}}$	The set of k th entrance protection grids at time step τ
$A_{i,\tau}^{\text{Entry}}$	Entry-oriented reachable area
$A_{i,\tau}^{\text{Exit}}$	Exit-oriented reachable area
$A_{i,\tau}^{\text{Reachable}}$	Reachable area
$A_{i,\tau}^{\text{AE},(m,n)}$	The exit-oriented reachable area where grid (m, n) is taken as the exit grid
$E_{i,\tau}^{\text{Generated}}$	The set of generated directed edges between the adjacent time step planes $(\tau - 1)$ and τ
E_i^{AE}	The set of edges that connects the destination node and available exit grids
g_i^{Entry}	The grids that UAV i 's entry point locates
g_i^{Exit}	The grids that UAV i 's exit point locates
$g_{(m,n,\tau)}$	Grid (m, n, τ)
$c_{(m,n,\tau)}$	The centre position of grid (m, n, τ)
$T_i^{\text{Rerouting}}$	UAV i 's optimal rerouting trajectory
$e_i^{(m,n,\tau)}_{(m',n',\tau+1)}$	The directed edge from point $c_{(m,n,\tau)}$ to point $c_{(m',n',\tau+1)}$
$c_{(m,n,\tau)}^{\text{Energy}}$	The energy consumption of edge $e_i^{(m,n,\tau)}_{(m',n',\tau+1)}$
$p_i(t)$	The position of UAV i at time t
p_i^τ	The position of UAV i at time step τ
p_i^{Entry}	The position of UAV i at the entry point position
p_i^{Exit}	The position of UAV i at the exit point position
(x_i^t, y_i^t)	The horizontal and vertical coordinates of UAV i at time t
$(x_i^{\text{Entry}}, y_i^{\text{Entry}})$	The coordinates of UAV i 's entry point
$(x_i^{\text{Exit}}, y_i^{\text{Exit}})$	The coordinates of UAV i 's exit point
$(m_i^{\text{Entry}}, n_i^{\text{Entry}})$	The grid that UAV i 's entry point locates
$(m_i^{\text{Exit}}, n_i^{\text{Exit}})$	The grid that UAV i 's exit point locates
$p_i^{(m,n)}$	The occupying rate of UAV i in grid (m, n)
$p_i^{(m,n,\tau)}$	UAV i 's occupying rate to grid (m, n, τ)
$p_{(\geq 2)}^{(m,n,\tau)}$	The probability of no less than 2 UAVs in grid (m, n, τ)
$p_{(0)}^{(m,n,\tau)}$	The probabilities of none UAV appearing in grid (m, n, τ)
$p_{(1)}^{(m,n,\tau)}$	The probabilities of one UAV appearing in grid (m, n, τ)
p^{Safety}	Safety threshold
$P_i^{\text{Min,Require}}$	UAV i 's minimum required power
$p_{\text{New}}^{(m,n,\tau)}$	The new UAV's occupying rate to grid (m, n, τ)
$r^{(m,n,\tau)}$	The maximum remaining occupiable rate
T_i	The set of time steps of UAV i flying within the airspace unit
τ_i^{Entry}	The time step of UAV i entering the airspace unit
τ_i^{Exit}	The time step of UAV i exiting the airspace unit
t^τ	The time of time step τ
t_i^{Entry}	The time of UAV i 's entry time
t_i^{Exit}	The time of UAV i 's exit time
Φ	Occupying rate identification threshold
ΔT	Time step length
φ	The weighted function of the edge
V_i^{max}	The maximum speed of UAV i
V_i^{min}	The minimum speed of UAV i
$ \pi $	The distance of two points
$N_i^{\text{Candidate}}$	The set of candidate nodes used to generate the directed graph for UAV i

(continued on next page)

Table A.6 (continued).

Symbols	Definitions
K	The set of layers of the entrance protection grids
k_{\max}	The maximum number of layers of the entrance protection grids
$I^{(m,n,\tau)}$	The set of UAVs whose probability of occupying grid (m, n, τ) is greater than or equal to Φ
n^{UAV}	Number of UAVs in the scenario
n^{IniCon}	Number of initial conflicts
$c_{\text{ExtEne}}^{\text{Tot}}$	Total extra energy consumption
$c_{\text{IniEne}}^{\text{Tot}}$	Total initial energy consumption
$t_{\text{Del}}^{\text{Tot}}$	Total delay time
$t_{\text{IniflTim}}^{\text{Tot}}$	Total initial flight time
$t_{\text{Hov}}^{\text{Tot}}$	Total hover time
$t_{\text{Com}}^{\text{Tot}}$	Total computing time

Appendix D. Iterative update of grid occupied rate

Calculating Eq. (5) is complex and time-consuming, especially with a large number of UAVs in the airspace unit. Therefore, we use iterative calculation to solve this problem. We introduce the remaining occupiable rate $r^{(m,n,\tau)}$, which is defined as the new UAV's maximum available occupying rate to grid (m, n, τ) satisfying the safety constraint. Given the current $p_{(1)}^{(m,n,\tau)}$ and $p_{(\geq 2)}^{(m,n,\tau)}$, if the new UAV's occupying rate to grid (m, n, τ) is $p_{\text{New}}^{(m,n,\tau)}$ and its current planned trajectory is conflict-free, we can get the following inequation:

$$p_{(\geq 2)}^{(m,n,\tau)} + p_{(1)}^{(m,n,\tau)} \times p_{\text{New}}^{(m,n,\tau)} \leq P^{\text{Safety}} \quad (\text{D.1})$$

Therefore, the value of the remaining occupiable rate $r^{(m,n,\tau)}$ can be represented as:

$$r^{(m,n,\tau)} = \min\{1, \frac{P^{\text{Safety}} - p_{(\geq 2)}^{(m,n,\tau)}}{p_{(1)}^{(m,n,\tau)}}\} \quad (\text{D.2})$$

The safety constraint (Eq. (1)) also can be represented as:

$$p_{\text{New}}^{(m,n,\tau)} \leq r^{(m,n,\tau)} \quad (\text{D.3})$$

We use the following iterative process to calculate the remaining occupiable rate $r^{(m,n,\tau)}$ and determine whether the grid satisfies the safety constraints:

1. When no UAV occupies grid (m, n, τ) , $p_{(0)}^{(m,n,\tau)} = 1$, $p_{(1)}^{(m,n,\tau)} = p_{(\geq 2)}^{(m,n,\tau)} = 0$ and $r^{(m,n,\tau)} = 1$.
2. When a new UAV occupies (m, n) with the occupying rate of $p_{\text{New}}^{(m,n,\tau)}$, $p_{(0)}^{(m,n,\tau)}$, $p_{(\geq 2)}^{(m,n,\tau)}$ and $p_{(1)}^{(m,n,\tau)}$ are updated in sequence as following:

$$p_{(0)}^{(m,n,\tau)} \leftarrow p_{(0)}^{(m,n,\tau)} \times (1 - p_{\text{New}}^{(m,n,\tau)}) \quad (\text{D.4})$$

$$p_{(\geq 2)}^{(m,n,\tau)} \leftarrow p_{(\geq 2)}^{(m,n,\tau)} + p_{(1)}^{(m,n,\tau)} \times p_{\text{New}}^{(m,n,\tau)} \quad (\text{D.5})$$

$$p_{(1)}^{(m,n,\tau)} \leftarrow 1 - p_{(0)}^{(m,n,\tau)} - p_{(\geq 2)}^{(m,n,\tau)} \quad (\text{D.6})$$

3. The remaining occupiable rate $r^{(m,n,\tau)}$ is calculated by Eq. (D.2).
4. Whether the grid satisfies the safety constraint can be judged by Eq. (D.3).
5. The steps 2–4 are repeated.

Appendix E. Method for identifying feasible rerouting points

In Fig. 7, sub-figure (1) shows the way to find reachable grids. The constraints of UAV i based flight speed performance can be represented as:

$$\left| \overline{p_i^{\text{Entry}} p_i^\tau} \right| \leq (t^\tau - t_i^{\text{Entry}}) V_i^{\max} \quad (\text{E.1})$$

$$\left| \overline{p_i^\tau p_i^{\text{Exit}}} \right| \leq (t_i^{\text{Exit}} - t^\tau) V_i^{\max} \quad (\text{E.2})$$

p_i^τ , p_i^{Entry} and p_i^{Exit} represent the position of UAV i at time step τ , the entry point position and the exit point position, respectively. t^τ , t_i^{Entry} and t_i^{Exit} represent the time of time step τ , UAV i 's entry time and exit time, respectively. V_i^{\max} represents the maximum speed of UAV i and $|\cdot|$ represents the distance of two points. Therefore, the reachable grid can be represented as:

$$\mathbf{G}_{i,\tau}^{\text{Reachable}} = \left\{ \mathbf{g}_{(m,n,\tau)} \mid \left| \overline{p_i^{\text{Entry}} c_{(m,n,\tau)}} \right| \leq (t^\tau - t_i^{\text{Entry}}) V_i^{\max}, \left| \overline{c_{(m,n,\tau)} p_i^{\text{Exit}}} \right| \leq (t_i^{\text{Exit}} - t^\tau) V_i^{\max}, \mathbf{g}_{(m,n,\tau)} \in \mathbf{G}_\tau \right\} \quad (\text{E.3})$$

where $\mathbf{G}_{i,\tau}^{\text{Reachable}}$ represents the set of reachable grids of UAV i at time step τ , \mathbf{G}_τ represents the set of grids at time step plane τ in the airspace unit, $\mathbf{g}_{(m,n,\tau)}$ represents grid (m, n, τ) and $\mathbf{c}_{(m,n,\tau)}$ represents the centre position of grid (m, n, τ) . Actually, according to the space-time prism theory (Siqi et al., 2018), the reachable constraints for the entry and exit points can be represented by two circles, respectively:

$$(x - x_i^{\text{Entry}})^2 + (y - y_i^{\text{Entry}})^2 \leq [(t^\tau - t^{\text{Entry}}) V_i^{\max}]^2 \quad (\text{E.4})$$

$$(x - x_i^{\text{Exit}})^2 + (y - y_i^{\text{Exit}})^2 \leq [(t^\tau - t^{\text{Exit}}) V_i^{\max}]^2 \quad (\text{E.5})$$

where $(x_i^{\text{Entry}}, y_i^{\text{Entry}})$ and $(x_i^{\text{Exit}}, y_i^{\text{Exit}})$ are the coordinates of UAV i 's entry and exit points, respectively. If we use $\mathbf{A}_{i,\tau}^{\text{Entry}}$ and $\mathbf{A}_{i,\tau}^{\text{Exit}}$ to represent the two circle areas (i.e., entry-oriented and exit-oriented reachable area, respectively), the reachable area $\mathbf{A}_{i,\tau}^{\text{Reachable}}$ can be represented as:

$$\mathbf{A}_{i,\tau}^{\text{Reachable}} = \mathbf{A}_{i,\tau}^{\text{Entry}} \cap \mathbf{A}_{i,\tau}^{\text{Exit}} \quad (\text{E.6})$$

Furthermore, since this paper allows the UAV to exit from an alternative exit point, Eq. (E.6) should be re-written as:

$$\mathbf{A}_{i,\tau}^{\text{Reachable}} = \mathbf{A}_{i,\tau}^{\text{Entry}} \cap \left(\mathbf{A}_{i,\tau}^{\text{Exit}} \cup \bigcup_{(m,n) \in \mathbf{G}_i^{\text{AE}}} \mathbf{A}_{i,\tau}^{\text{AE},(m,n)} \right) \quad (\text{E.7})$$

where $\mathbf{A}_{i,\tau}^{\text{AE},(m,n)}$ represents the exit-oriented reachable area where grid (m, n) is taken as the exit grid, and \mathbf{G}_i^{AE} represents the set of UAV i 's alternative exit grids.

Therefore, Eq. (E.3) can also be represented as:

$$\mathbf{G}_{i,\tau}^{\text{Reachable}} = \left\{ \mathbf{g}_{(m,n,\tau)} \mid \mathbf{c}_{(m,n,\tau)} \in \mathbf{A}_{i,\tau}^{\text{Reachable}}, \mathbf{g}_{(m,n,\tau)} \in \mathbf{G}_\tau \right\} \quad (\text{E.8})$$

For reachable grid (m, n, τ) , assume that UAV i is in the grid, based on UAV i 's occupying rates to grids in this case (refer to sub-figure (2.1), (2.2) or (2.3)) and the current grids' occupied rates (refer to sub-figure (3)), and then determine whether the safety constraint is satisfied by the conflict detection method mentioned in Section 3.2 (especially refer to Fig. 6). If the safety constraint is satisfied, reachable grid (m, n, τ) is an available grid; otherwise, it is an unavailable grid (e.g., sub-figure (4)). Therefore, the sets of available and unavailable grids for UAV i at time step τ are, respectively, represented as:

$$\mathbf{G}_{i,\tau}^{\text{Available}} = \left\{ \mathbf{g}_{(m,n,\tau)} \mid (x_i^\tau, y_i^\tau) = \mathbf{c}_{(m,n,\tau)}, \forall \mathbf{g}_{(m',n',\tau)} \in \mathbf{G}_\tau : p_{i,\tau}^{(m',n',\tau)} \leq r_{\tau}^{(m',n',\tau)}, \mathbf{g}_{(m,n,\tau)} \in \mathbf{G}_{i,\tau}^{\text{Reachable}} \right\} \quad (\text{E.9})$$

$$\mathbf{G}_{i,\tau}^{\text{Unavailable}} = \complement_{\mathbf{G}_{i,\tau}^{\text{Reachable}}} \mathbf{G}_{i,\tau}^{\text{Available}} \quad (\text{E.10})$$

where \complement is the complement symbol. All reachable grids are judged in turn to obtain all available grids at the plane of time step τ .

Since the interval ΔT of time steps is very small (e.g., 1 s) in this paper, we assume that the entry and exit points of the UAV are the centre points of a grid located at the edge of the airspace unit, and their corresponding CTAs are just at a time step. We use $N_i^{\text{Candidate}}$ to represent the set of candidate nodes used to generate the directed graph for UAV i , and for the sake of formulation, we let:

$$\mathbf{G}_{i,\tau_i^{\text{Entry}}}^{\text{Available}} = \left\{ \mathbf{g}_i^{\text{Entry}} \right\} \quad (\text{E.11})$$

$$\mathbf{G}_{i,\tau_i^{\text{Exit}}}^{\text{Available}} = \left\{ \mathbf{g}_i^{\text{Exit}} \right\} \quad (\text{E.12})$$

where τ_i^{Entry} and τ_i^{Exit} are the time steps of UAV i entering and exiting the airspace unit, respectively, and $\mathbf{g}_i^{\text{Entry}}$ and $\mathbf{g}_i^{\text{Exit}}$ are the grids that UAV i 's entry and exit points locate, respectively. Therefore, $N_i^{\text{Candidate}}$ can be represented as:

$$N_i^{\text{Candidate}} = \left\{ \mathbf{c}_{(m,n,\tau)} \mid \mathbf{g}_{(m,n,\tau)} \in \mathbf{G}_{i,\tau}^{\text{Available}}, \tau \in T_i \right\} \quad (\text{E.13})$$

where T_i is the set of time steps of UAV i flying within the airspace unit.

Appendix F. Energy consumption model for EHang 184

Since the UAV is assumed to be moving uniformly in a straight line of a horizontal plane, the UAV force analysis is shown in Fig. F.20. Referring to the state-of-the-art UAV energy consumption study on EHang 184 (Pradeep and Wei, 2019), the formula for calculating the UAV power required is derived as follows, where the nomenclature and the preset value of the parameters are referred to Table F.7:

$$T = \sqrt{G^2 + D^2} \quad (\text{F.1})$$

$$D = \frac{\rho V^2 C_D F}{2} \quad (\text{F.2})$$

$$G = mg \quad (\text{F.3})$$

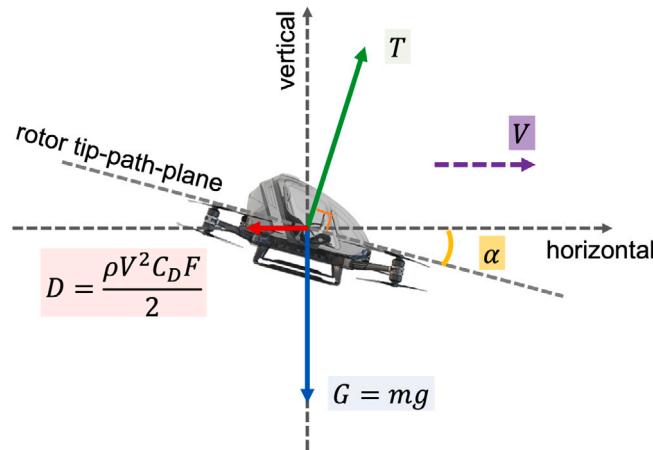


Fig. F.20. UAV force analysis diagram.

Table F.7

Nomenclature for the energy consumption model.

Symbols	Meanings	Preset values
T	Thrust produced by all rotors	N/A
D	Aerodynamic drag	N/A
ρ	Air density	1.205 kg/m ³
V	True airspeed	N/A
C_D	Drag coefficient	1
F	Equivalent front plate area	2.11 m ²
G	Gravity	N/A
m	UAV mass	240 kg
g	Gravitational acceleration	9.80 m/s ²
T_{Rotor}	Thrust produced by a rotor	N/A
N_{Rotor}	Number of rotors	8
A	Rotor disk area	2.01 m ²
v_h	Induced velocity in hover	N/A
v_i	Induced velocity in forward flight	N/A
α	Angle of attack	N/A
$P_{\text{Rotor}}^{\text{Induced}}$	Induced power loss of an isolated rotor in forward flight	N/A
P_{Arm}	Induced power loss of an arm	N/A
χ	Rotor interference factor for the coaxial rotor system	1.0
P_{Required}	Power required in forward flight	N/A
N_{Arm}	Number of arms	4

$$T_{\text{Rotor}} = \frac{T}{N_{\text{Rotor}}} \quad (\text{F.4})$$

$$v_h = \sqrt{\frac{T_{\text{Rotor}}}{2\rho A}} \quad (\text{F.5})$$

$$v_i = \frac{v_h^2}{\sqrt{(V \cos \alpha)^2 + (V \sin \alpha + v_i)^2}} \quad (\text{F.6})$$

$$P_{\text{Rotor}}^{\text{Induced}} = T_{\text{Rotor}} v_i \quad (\text{F.7})$$

$$P_{\text{Arm}} = 2 P_{\text{Rotor}}^{\text{Induced}} (1 + \chi) \quad (\text{F.8})$$

$$P_{\text{Required}} = N_{\text{Arm}} P_{\text{Arm}} + TV \sin \alpha \quad (\text{F.9})$$

It should be noted that since the accurate energy consumption model is beyond the scope of this paper, the above derivation process is reasonably simplified, and the obtained energy consumption model is an approximation of the real situation, which contains the following assumptions referring to the relevant studies (Pradeep and Wei, 2019):

1. The net drag on the UAV is assumed to be equivalent to the drag on the fuselage of the aircraft

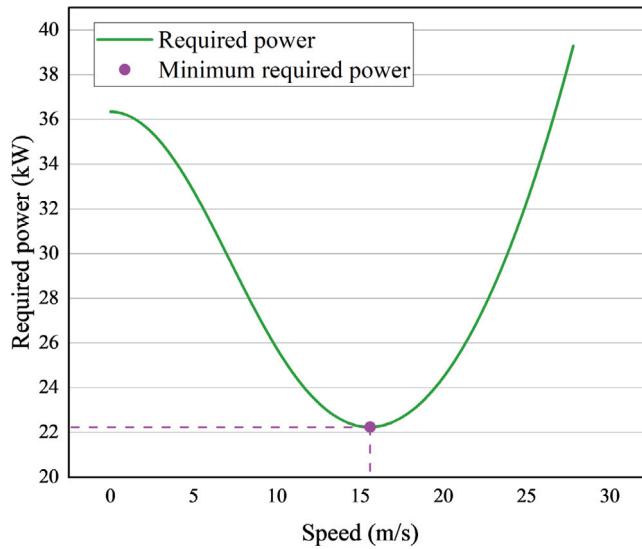


Fig. F.21. Speed - required power curve of EHang 184.

2. Since the effect of wind is not considered, and the vertical velocity is zero, the bank angle is equal to the angle of attack.
3. Each rotor provides the same thrust.
4. Each arm has the same induced power loss.
5. The profile drag is assumed to be constant in magnitude and hence has a negligible impact on the variation of the instantaneous power required. Therefore, the required power equals the sum of induced power and parasite power.
6. The drag coefficient is assumed to be a constant (i.e., 1).
7. The interference factor χ for all the rotors is assumed to be 1.0.

The relationship between cruising speed and the required power of EHang 184 can be obtained through simulation, as shown in Fig. F.21. The minimum power required is 22.24 kW, obtained at a speed of 15.6 m/s. It should be noted that since Eq. (F.6) is a transcendental equation about v_i , where the solution speed is slow, in practice, the speed and required power comparison table can be used to obtain the required power corresponding to the speed quickly.

Appendix G. Occupying rate distribution nominal map

Central and compact occupying rate distribution nominal maps are introduced to avoid repeated integral calculation in cases where the UAV is and is not located at the grid's centre point, respectively. Fig. G.22 illustrates how occupying rate distribution nominal maps are constructed and used.

The central occupying rate distribution nominal map for UAV i is obtained by the integral calculation described in Appendix C, which is a $N_i^X \times N_i^Y$ matrix, represented by P_i^{CenM} . The elements in the matrix are the occupying rates to the grids when UAV i is in grid $(0, 0)$. N_i^X and N_i^Y are the maximum range of the grid occupied by UAV i in the X and Y directions, respectively (i.e. the occupying rate greater than the occupying rate identification threshold Φ). Since the position probability distribution of the UAV in this paper is cross-symmetric with respect to the central point (refer to Appendix B), the nominal map requires only one quadrant of occupying rate distribution information.

On the basis of the central occupying rate distribution nominal map, the compact occupying rate distribution nominal map for UAV i is calculated based on the worst case. When calculating UAV i 's occupying rate to grid (m, n) , it is assumed that UAV i 's planned trajectory point is located closest to grid (m, n) in the grid in which it is located (i.e., (m_i^L, n_i^L)). Therefore, we first construct occupying rate distribution maps (central, top, right and top-right), where the UAV's planned trajectory point is assumed to be located at $(\frac{1}{2}L^G, \frac{1}{2}L^G)$, $(\frac{1}{2}L^G, L^G)$, $(L^G, \frac{1}{2}L^G)$ and (L^G, L^G) , respectively, and the corresponding matrices are represented by P_i^C , P_i^T , P_i^R and P_i^{TR} , respectively. L^G is the grid side length. It should be noted that the occupying rate distribution map (central) is the same as the central occupying rate distribution nominal map. Therefore, the compact occupying rate distribution nominal map P_i^{ComM} can be represented as:

$$P_i^{\text{ComM}}(x, y) = \begin{cases} P_i^{TR}(x, y) & x \geq 1, y \geq 1 \\ P_i^T(x, y) & x = 0, y \geq 1 \\ P_i^R(x, y) & x \geq 1, y = 0 \\ P_i^C(x, y) & x = 0, y = 0 \end{cases}, \quad (G.1)$$

$$x = 0, 1, \dots, N_i^X - 1, y = 0, 1, \dots, N_i^Y - 1$$

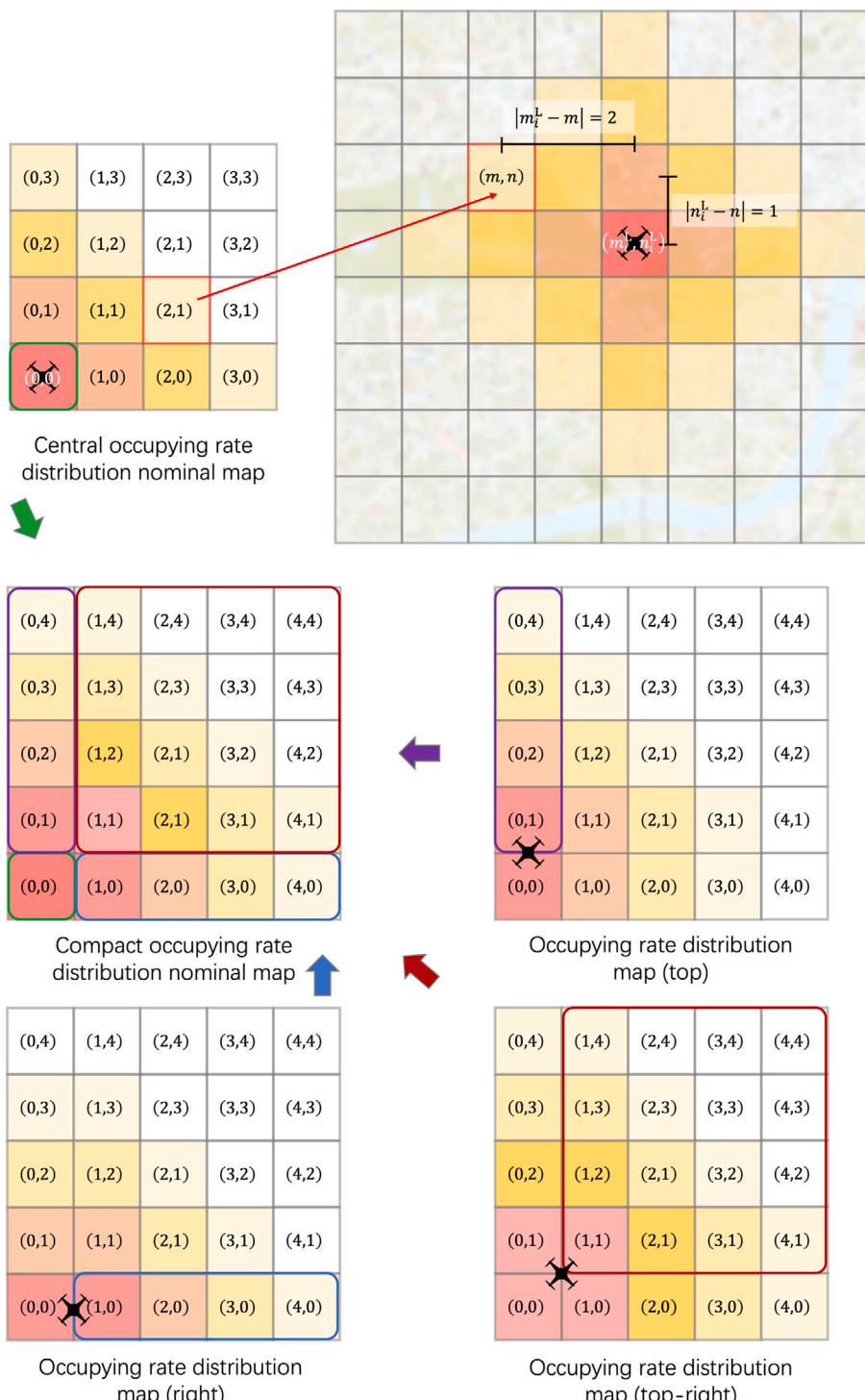


Fig. G.22. Calculation method of central and compact nominal maps.

Therefore, based on the central or compact occupying rate distribution nominal map (both presented by P_i^{Map}), UAV i 's occupying rate to grid (m, n) (i.e., $p_i^{(m,n)}$) can be obtained by:

$$p_i^{(m,n)} = \begin{cases} P_i^{\text{Map}} \left(|m_i^L - m|, |n_i^L - n| \right) & |m_i^L - m| < N_i^X, |n_i^L - n| < N_i^Y \\ 0 & \text{otherwise} \end{cases} \quad (\text{G.2})$$

where $P_i^{\text{Map}}(x, y)$ represents the value at position (x, y) in the matrix P_i^{CenM} or P_i^{ComM} .

Appendix H. Planned trajectory generation

The planned trajectory generation steps are as follows:

1. Entry and exit grids are randomly selected, and their centre points are respectively the entry point and the exit point, i.e., $(x_i^{\text{Entry}}, y_i^{\text{Entry}})$ and $(x_i^{\text{Exit}}, y_i^{\text{Exit}})$.
2. A time step is randomly generated within the simulation duration, whose corresponding time is taken as the entry time, i.e., τ_i^{Entry} .
3. A time step that meets the flight performance of the UAV (i.e., Eq. (H.1)) is randomly generated, whose corresponding time is taken as the exit time, i.e., τ_i^{Exit} .

$$V_{\text{Planned}}^{\min} \leq \frac{\sqrt{(x_i^{\text{Exit}} - x_i^{\text{Entry}})^2 + (y_i^{\text{Exit}} - y_i^{\text{Entry}})^2}}{\tau_i^{\text{Exit}} - \tau_i^{\text{Entry}}} \leq V_{\text{Planned}}^{\max} \quad (\text{H.1})$$

4. The above steps are repeated by N times (equal to traffic density times simulation duration), and finally, the planned trajectories of N UAVs are obtained, i.e., $(x_i^{\text{Entry}}, y_i^{\text{Entry}}, \tau_i^{\text{Entry}}), (x_i^{\text{Exit}}, y_i^{\text{Exit}}, \tau_i^{\text{Exit}})$, $i = 1, 2, \dots, N$.

Appendix I. Safety threshold value setting

Because this paper assumes that all UAVs have the same positioning error level, all UAVs have the same grid occupancy distribution if the grid where the UAVs reside is taken as a reference. Based on the parameters instantiated in this paper, the UAV's occupying rates are shown in Fig. 11. Since safety levels based on grid occupancy need to be systematically explored in the future, this paper only discusses simple cases. If it is considered safe as long as two UAVs do not appear in the same spatio-temporal grid, the safety threshold value can be obtained by Eq. (I.1), assuming that the UAV is located in the centre of grid (m, n) .

$$P^{\text{Safety}} = p_i^{(m,n)} \times p_i^{(m,n)} \quad (\text{I.1})$$

More conservatively, if two UAVs in adjacent grids are considered unsafe, the safety threshold value can be obtained by Eq. (I.2).

$$P^{\text{Safety}} = p_i^{(m,n)} \times p_i^{(m,n+1)} \quad (\text{I.2})$$

In general, if we want two UAVs to be at least $N^{\text{Separation}}$ grids apart, the safety threshold value can be obtained by Eq. (I.3).

$$P^{\text{Safety}} = p_i^{(m,n)} \times p_i^{(m,n+N^{\text{Separation}})} \quad (\text{I.3})$$

In the instantiation of this paper, $N^{\text{Separation}} = 1$, i.e., $P^{\text{Safety}} = 0.2111 \times 0.1089 \approx 0.0230$.

References

- Ait-Saadi, A., Meraïhi, Y., Soukane, A., Ramdane-Cherif, A., Gabis, A.B., 2022. A novel hybrid chaotic aquila optimization algorithm with simulated annealing for unmanned aerial vehicles path planning. *Comput. Electr. Eng.* 104, 108461.
- Alejo, D., Cobano, J.A., Heredia, G., Ollero, A., 2016. An efficient method for multi-UAV conflict detection and resolution under uncertainties. In: Robot 2015: Second Iberian Robotics Conference: Advances in Robotics, Volume 1. Springer, pp. 635–647.
- Bertram, J., Wei, P., Zambreno, J., 2022. A fast Markov decision process-based algorithm for collision avoidance in urban air mobility. *IEEE Trans. Intell. Transp. Syst.* 23 (9), 15420–15433.
- Bertram, J., Zambreno, J., Wei, P., 2023. Efficient unmanned aerial systems navigation with collision avoidance in dense urban environments. *IEEE Trans. Intell. Transp. Syst.*
- Blom, H.A., Jiang, C., Grimme, W.B., Mitici, M., Cheung, Y.S., 2021. Third party risk modelling of unmanned aircraft system operations, with application to parcel delivery service. *Reliab. Eng. Syst. Saf.* 214, 107788.
- Brittain, M., Wei, P., 2022. Scalable autonomous separation assurance with heterogeneous multi-agent reinforcement learning. *IEEE Trans. Autom. Sci. Eng.* 19 (4), 2837–2848.
- Catapult, 2021. Towards a UTM system for the UK preparing the UK for the commercial drone industry. URL <https://cp.catapult.org.uk/wp-content/uploads/2021/01/Towards-a-UTM-System-for-the-UK.pdf>.
- Chai, R., Tsourdos, A., Savvaris, A., Wang, S., Xia, Y., Chai, S., 2020. Fast generation of chance-constrained flight trajectory for unmanned vehicles. *IEEE Trans. Aerosp. Electron. Syst.* 57 (2), 1028–1045.
- Chen, Y., Hu, M., Yang, L., 2022. Autonomous planning of optimal four-dimensional trajectory for real-time en-route airspace operation with solution space visualisation. *Transp. Res. C* 140, 103701.
- Chen, Y., Yang, D., Yu, J., 2018. Multi-UAV task assignment with parameter and time-sensitive uncertainties using modified two-part wolf pack search algorithm. *IEEE Trans. Aerosp. Electron. Syst.* 54 (6), 2853–2872.
- Cheng, C., Adulyasak, Y., Rousseau, L.-M., 2020. Drone routing with energy function: Formulation and exact algorithm. *Transp. Res. B* 139, 364–387.

- Courchelle, V., Soler, M., González-Arribas, D., Delahaye, D., 2019. A simulated annealing approach to 3D strategic aircraft deconfliction based on en-route speed changes under wind and temperature uncertainties. *Transp. Res. C* 103, 194–210.
- Dai, W., Pang, B., Low, K.H., 2021. Conflict-free four-dimensional path planning for urban air mobility considering airspace occupancy. *Aerosp. Sci. Technol.* 119, 107154.
- Dias, F.H., Hijazi, H., Rey, D., 2022. Disjunctive linear separation conditions and mixed-integer formulations for aircraft conflict resolution. *European J. Oper. Res.* 296 (2), 520–538.
- Dias, F.H., Rey, D., 2022. Robust aircraft conflict resolution under trajectory prediction uncertainty. *Oper. Res. Lett.* 50 (5), 503–508.
- Ding, W., Zhang, Y., Hansen, M., 2018. Downstream impact of flight rerouting. *Transp. Res. C* 88, 176–186.
- Du, B., Chen, J., Sun, D., Manyam, S.G., Casbeer, D.W., 2021. UAV trajectory planning with probabilistic geo-fence via iterative chance-constrained optimization. *IEEE Trans. Intell. Transp. Syst.* 23 (6), 5859–5870.
- EASA, 2020. High-level regulatory framework for the U-space.
- Erke, S., Bin, D., Yiming, N., Qi, Z., Liang, X., Dawei, Z., 2020. An improved A-star based path planning algorithm for autonomous land vehicles. *Int. J. Adv. Robot. Syst.* 17 (5), 1729881420962263.
- FAA, 2023. UTM concept of operations version 2.0.
- Freitas, H., Faical, B.S., e Silva, A.V.C., Ueyama, J., 2020. Use of UAVs for an efficient capsule distribution and smart path planning for biological pest control. *Comput. Electron. Agric.* 173, 105387.
- Gaxiola, C.A.N., Barrado, C., Royo, P., Pastor, E., 2018. Assessment of the north European free route airspace deployment. *J. Air Transp. Manag.* 73, 113–119.
- González-Arribas, D., Soler, M., Sanjurjo-Rivo, M., 2018. Robust aircraft trajectory planning under wind uncertainty using optimal control. *J. Guid. Control Dyn.* 41 (3), 673–688.
- Guo, J., Xia, W., Hu, X., Ma, H., 2022. Feedback RRT* algorithm for UAV path planning in a hostile environment. *Comput. Ind. Eng.* 174, 108771.
- Hart, P.E., Nilsson, N.J., Raphael, B., 1968. A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans. Syst. Sci. Cybern.* 4 (2), 100–107.
- Ho, F., Geraldes, R., Gonçalves, A., Cavazza, M., Prendinger, H., 2018. Improved conflict detection and resolution for service UAVs in shared airspace. *IEEE Trans. Veh. Technol.* 68 (2), 1231–1242.
- Ho, F., Geraldes, R., Goncalves, A., Rigault, B., Ooseda, A., Cavazza, M., Prendinger, H., 2019. Pre-flight conflict detection and resolution for UAV integration in shared airspace: Sendai 2030 model case. *IEEE Access* 7, 170226–170237.
- Hong, Y., Choi, B., Oh, G., Lee, K., Kim, Y., 2017. Nonlinear conflict resolution and flow management using particle swarm optimization. *IEEE Trans. Intell. Transp. Syst.* 18 (12), 3378–3387.
- Hu, J., Wang, M., Zhao, C., Pan, Q., Du, C., 2020. Formation control and collision avoidance for multi-UAV systems based on voronoi partition. *Sci. China Technol. Sci.* 63 (1), 65–72.
- Huang, M., Ochieng, W.Y., Macias, J.J.E., Ding, Y., 2021. Accuracy evaluation of a new generic trajectory prediction model for unmanned aerial vehicles. *Aerosp. Sci. Technol.* 119, 107160.
- Jenie, Y.I., van Kampen, E.J., Ellerbroek, J., Hoekstra, J.M., 2017. Safety assessment of a UAV CD&R system in high density airspace using Monte Carlo simulations. *IEEE Trans. Intell. Transp. Syst.* 19 (8), 2686–2695.
- Johnson, W., 2013. Rotorcraft Aeromechanics, Vol. 36, Cambridge University Press.
- Kleinbekman, I.C., Mitici, M., Wei, P., 2020. Rolling-horizon electric vertical takeoff and landing arrival scheduling for on-demand urban air mobility. *J. Aerosp. Inf. Syst.* 17 (3), 150–159.
- Kochenderfer, M.J., Holland, J.E., Chryssanthacopoulos, J.P., 2012. Next generation airborne collision avoidance system. *Linc. Lab. J.* 19 (1), 17–33.
- Li, A., Hansen, M., Zou, B., 2022. Traffic management and resource allocation for UAV-based parcel delivery in low-altitude urban space. *Transp. Res. C* 143, 103808.
- Livadas, C., Lygeros, J., Lynch, N.A., 2000. High-level modeling and analysis of the traffic alert and collision avoidance system (TCAS). *Proc. IEEE* 88 (7), 926–948.
- Luis, C.E., Vukoslavljev, M., Schoellig, A.P., 2020. Online trajectory generation with distributed model predictive control for multi-robot motion planning. *IEEE Robot. Autom. Lett.* 5 (2), 604–611.
- Mellinger, D., Kushleyev, A., Kumar, V., 2012. Mixed-integer quadratic program trajectory generation for heterogeneous quadrotor teams. In: 2012 IEEE International Conference on Robotics and Automation. IEEE, pp. 477–483.
- Ning, Q., Tao, G., Chen, B., Lei, Y., Yan, H., Zhao, C., 2019. Multi-UAVs trajectory and mission cooperative planning based on the Markov model. *Phys. Commun.* 35, 100717.
- Ong, H.Y., Kochenderfer, M.J., 2017. Markov decision process-based distributed conflict resolution for drone air traffic management. *J. Guid. Control Dyn.* 40 (1), 69–80.
- Pang, B., Low, K.H., Duong, V.N., 2024. Chance-constrained UAM traffic flow optimization with fast disruption recovery under uncertain waypoint occupancy time. *Transp. Res. C* 161, 104547.
- Pang, B., Low, K.H., Lv, C., 2022. Adaptive conflict resolution for multi-UAV 4D routes optimization using stochastic fractal search algorithm. *Transp. Res. C* 139, 103666.
- Peng, N., Xi, Y., Rao, J., Ma, X., Ren, F., 2021. Urban multiple route planning model using dynamic programming in reinforcement learning. *IEEE Trans. Intell. Transp. Syst.* 23 (7), 8037–8047.
- Penin, B., Giordano, P.R., Chaumette, F., 2018. Minimum-time trajectory planning under intermittent measurements. *IEEE Robot. Autom. Lett.* 4 (1), 153–160.
- Perez-Leon, H., Acevedo, J.J., Maza, I., Ollero, A., 2021. Integration of a 4D-trajectory follower to improve multi-UAV conflict management within the U-space context. *J. Intell. Robot. Syst.* 102 (3), 62.
- Pradeep, P., Wei, P., 2019. Energy-efficient arrival with RTA constraint for multirotor eVTOL in urban air mobility. *J. Aerosp. Inf. Syst.* 16 (7), 263–277.
- PwC, 2018. Skies without limits. URL <https://www.pwc.co.uk/intelligent-digital/drones/Drones-impact-on-the-UK-economy-FINAL.pdf>.
- Roberge, V., Tarbouchi, M., Labonté, G., 2012. Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning. *IEEE Trans. Ind. Inform.* 9 (1), 132–141.
- Seo, J., Kim, Y., Kim, S., Tsourdos, A., 2017. Collision avoidance strategies for unmanned aerial vehicles in formation flight. *IEEE Trans. Aerosp. Electron. Syst.* 53 (6), 2718–2734.
- Seyedipour, S., Nobahari, H., Prandini, M., 2022. A probabilistic approach to mid-term conflict detection with accuracy estimate. *J. Franklin Inst.* 359 (16), 9193–9219.
- Silvagni, M., Tonoli, A., Zenerino, E., Chiaberge, M., 2017. Multipurpose UAV for search and rescue operations in mountain avalanche events. *Geomatics Nat. Hazards Risk* 8 (1), 18–33.
- Siqi, H., Cheng, S., Zhang, Y., 2018. A multi-aircraft conflict detection and resolution method for 4-dimensional trajectory-based operation. *Chin. J. Aeronaut.* 31 (7), 1579–1593.
- SJU, 2018. Blueprint on U-space.
- Tan, Q., Wang, Z., Ong, Y.-S., Low, K.H., 2019. Evolutionary optimization-based mission planning for UAS traffic management (UTM). In: 2019 International Conference on Unmanned Aircraft Systems. ICUAS, IEEE, pp. 952–958.
- Tang, Y., Xu, Y., 2023. Incorporating optimization in strategic conflict resolution for UAS traffic management. *IEEE Trans. Intell. Transp. Syst.*

- Tang, Y., Xu, Y., Inalhan, G., 2021a. Incorporating optimisation in strategic conflict resolution service in u-space. In: SIDs2021: 11th SESAR Innovation Days. pp. 1–8.
- Tang, H., Zhang, Y., Mohmoodian, V., Charkhgard, H., 2021b. Automated flight planning of high-density urban air mobility. *Transp. Res. C* 131, 103324.
- Valenti Clari, M., Ruigrok, R., Hoekstra, J., 2001. Cost-benefit study of free flight with airborne separation assurance. In: AIAA Guidance, Navigation, and Control Conference and Exhibit. p. 4361.
- Wang, Z., Pan, W., Li, H., Wang, X., Zuo, Q., 2022. Review of deep reinforcement learning approaches for conflict resolution in air traffic control. *Aerospace* 9 (6), 294.
- Wilhelm, J.P., Clem, G., 2019. Vector field UAV guidance for path following and obstacle avoidance with minimal deviation. *J. Guid. Control Dyn.* 42 (8), 1848–1856.
- Wu, Y., 2021. A survey on population-based meta-heuristic algorithms for motion planning of aircraft. *Swarm Evol. Comput.* 62, 100844.
- Wu, X., Lei, Y., Tong, X., Zhang, Y., Li, H., Qiu, C., Guo, C., Sun, Y., Lai, G., 2022a. A non-rigid hierarchical discrete grid structure and its application to UAVs conflict detection and path planning. *IEEE Trans. Aerosp. Electron. Syst.* 58 (6), 5393–5411.
- Wu, Y., Wu, S., Hu, X., 2021. Multi-constrained cooperative path planning of multiple drones for persistent surveillance in urban environments. *Complex Intell. Syst.* 7, 1633–1647.
- Wu, P., Yang, X., Wei, P., Chen, J., 2022b. Safety assured online guidance with airborne separation for urban air mobility operations in uncertain environments. *IEEE Trans. Intell. Transp. Syst.* 23 (10), 19413–19427.
- Yang, X., Alvarez, L.M., Bruggemann, T., 2013. A 3D collision avoidance strategy for UAVs in a non-cooperative environment. *J. Intell. Robot. Syst.* 70, 315–327.
- Yang, X., Deng, L., Liu, J., Wei, P., Li, H., 2020. Multi-agent autonomous operations in urban air mobility with communication constraints. In: AIAA Scitech 2020 Forum. p. 1839.
- Yang, X., Wei, P., 2020. Scalable multi-agent computational guidance with separation assurance for autonomous urban air mobility. *J. Guid. Control Dyn.* 43 (8), 1473–1486.
- Yang, X., Wei, P., 2021. Autonomous free flight operations in urban air mobility with computational guidance and collision avoidance. *IEEE Trans. Intell. Transp. Syst.* 22 (9), 5962–5975.
- Yang, J., Xu, X., Yin, D., Ma, Z., Shen, L., 2019. A space mapping based 0–1 linear model for onboard conflict resolution of heterogeneous unmanned aerial vehicles. *IEEE Trans. Veh. Technol.* 68 (8), 7455–7465.
- Yao, W., Qi, N., Wan, N., Liu, Y., 2019. An iterative strategy for task assignment and path planning of distributed multiple unmanned aerial vehicles. *Aerosp. Sci. Technol.* 86, 455–464.
- Zammit, C., Van Kampen, E.-J., 2022. Comparison between A* and RRT algorithms for 3D UAV path planning. *Unmanned Syst.* 10 (02), 129–146.
- Zhang, Z., Li, J., Wang, J., 2018. Sequential convex programming for nonlinear optimal control problems in UAV path planning. *Aerosp. Sci. Technol.* 76, 280–290.
- Zhang, M., Yan, C., Dai, W., Xiang, X., Low, K.H., 2023. Tactical conflict resolution in urban airspace for unmanned aerial vehicles operations using attention-based deep reinforcement learning. *Green Energy Intell. Transp.* 2 (4), 100107.
- Zou, Y., Zhang, H., Zhong, G., Liu, H., Feng, D., 2021. Collision probability estimation for small unmanned aircraft systems. *Reliab. Eng. Syst. Saf.* 213, 107619.