RISK-BASED ROUTE PLANNING AND DECISION MAKING FOR UNMANNED AIRCRAFT SYSTEM OPERATIONS IN URBAN ENVIRONMENTS

BIZHAO PANG

SCHOOL OF MECHANICAL AND AEROSPACE ENGINEERING

NANYANG TECHNOLOGICAL UNIVERSITY

2023

Risk-Based Route Planning and Decision Making for Unmanned Aircraft System Operations in Urban Environments

BIZHAO PANG

School of Mechanical and Aerospace Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Doctor of Philosophy

2023

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

15/February/2023 庞笔照

. . . .

Date Bizhao Pang

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

04/July/2023

. . . .

Date   Professor Vu N. Duong

Authorship Attribution Statement

This thesis contains material from 4 papers published in the following peer-reviewed journal(s) /conferences from Chapters accepted at conferences in which I am listed as an author.

Chapter 3 is published as **Bizhao Pang**, Wei Dai, Thu Ra, and Kin Huat Low. "A concept of airspace configuration and operational rules for UAS in current airspace." In 2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC), pp. 1-9. IEEE, 2020. https://doi.org/10.1109/DASC50938.2020.9256627

The contributions of the co-authors are as follows:

- I was the lead author. I constructed the ideas, wrote the manuscript, and conducted all simulations and experiments.
- Wei Dai participated in the discussion of the paper framework and reviewed the manuscript.
- Thu Ra provided the platform for visualization.
- Prof. Low discussed the ideas, provided the computational resources, and reviewed the manuscript.

Chapter 3 is also published as **Bizhao Pang**, Mingcheng Zhang, Chao Deng, and Kin Huat Low. "Investigation of Flight Technical Error for UAV Separation Requirement Based on Flight Trajectory Data." In AIAA Aviation 2022 Forum, p. 3763. 2022. https://doi.org/10.2514/6.2022-3763

The contributions of the co-authors are as follows:

- I was the lead author. I constructed the ideas, wrote the manuscript, and conducted all simulations and experiments.
- Mingcheng Zhang participated in the discussion of the paper framework and partially assisted the flight experiments.
- Chao Deng partially assisted the flight experiments and data collection.
- Prof. Low revised the paper structured, provided the experiment resources (e.g., hardware and software of drones), and reviewed the manuscript.

Chapter 4 is published as **Bizhao Pang**, Xinting Hu, Wei Dai, and Kin Huat Low. "UAV path optimization with an integrated cost assessment model considering third-party risks in metropolitan environments." *Reliability Engineering & System Safety* 222 (2022): 108399. https://doi.org/10.1016/j.ress.2022.108399

The contributions of the co-authors are as follows:

- I was the lead author. I constructed the ideas, formulated the mathematical models, wrote and revised the manuscript, and conducted all simulations.
- Xinting Hu participated in the risk assessment modelling and edited the manuscript drafts.
- Wei Dai participated in the discussion and reviewed the manuscript.
- Prof. Low discussed the ideas and reviewed the manuscript.

Chapter 5 is published as **Bizhao Pang**, Kin Huat Low, and Chen Lv. "Adaptive conflict resolution for multi-UAV 4D routes optimization using stochastic fractal search algorithm." *Transportation Research Part C: Emerging Technologies* 139 (2022): 103666. https://doi.org/10.1016/j.trc.2022.103666

The contributions of the co-authors are as follows:

- I was the lead author. I constructed the ideas, formulated the mathematical models, wrote and revised the manuscript, and conducted all simulations.
- Prof. Low discussed the ideas and reviewed the manuscript.
- Prof. Lyu discussed the research ideas, revised the manuscript structure.

16/February/2023

. . . .

Date   Bizhao Pang

Acknowledgment

Doing PhD is one of the best choices I made in my life, as it offers me an opportunity to experience the most challenging but interesting research journey. Along the journey, I have gained substantial knowledge, invaluable friendship, and various experiences in life. However, I cannot achieve that without the tremendous help, support, and encouragement from my supervisors, colleagues, friends, and my family. I would sincerely like to thank each and every one of you.

First of all, I am deeply grateful to my supervisor Professor Vu N. Duong for taking care of me during hard times, guiding me through confusion times, and supporting me at all times. I am also very thankful to my co-supervisor Professor Lyu Chen. His broad knowledge and insightful advice on intelligent decision making have been instrumental in helping me complete this thesis. I would also like to thank Professor Low Kin Huat for his kind guidance, support, and encouragement for my PhD studies. I am so grateful to have them with me on my research journey and appreciate their great patience and support throughout the process.

My gratitude is also extended to my teachers, colleagues, and friends, who have supported me both professionally and personally. I would like to thank Professor Henk Blom, Professor Sameer Alam, Professor Wang Yifan, and Professor Mao Kezhi. The discussions with them helped me to jump out of the "box" and understand the research problem in a broader and deeper view. My thanks are also expressed to my teammates in ATMRI, NTU, especially to Ms Xinting Hu, Mr Wei Dai, Dr. Yu Wu, Dr John Wang, Ms Chao Deng, Mr Qingyu Tan, Mr Mingcheng Zhang, Mr T. Thanaraj, Mr Anush Kumar, and so on. Their constructive feedback, collaborative spirit, and encouragement have been a source of inspiration and motivation, which enriched my knowledge and research skills.

My sincere appreciation must also be conveyed to the anonymous journal and thesis reviewers for the research works in this thesis. The more than 300 comments given by them have fundamentally helped to improve this research work.

Last but the most important, I would like to thank my parents and elder sister. Their endless love, support, and understanding have always been so silent but powerful, which drive me towards the future.

Contents

List of Figures

List of Tables

Abstract

Unmanned aircraft system (UAS) has been fast developing by leveraging advanced technologies such as power and flight control systems. The applications of UAS in urban environments also show a significant increase, with prominent use cases such as parcel delivery, security patrol, aerial photography, and data collection, which significantly facilitate our daily life and public services. However, the rapid boom of UAS operations in and around urban airspace poses challenges for aviation regulatory bodies and brings various safety risks to the public. Specifically, drone operations may fail and crash into people, causing ground fatality risk. Airborne collisions between drones yield air risk, and the falling of collided drones causes knock-on ground risk.

These safety risks could result in severe casualties to third-party people and considerable property loss to urban infrastructures, which need immediate research efforts to assess and mitigate the safety risks. This research problem is further complicated due to various uncertainties, such as weather and population movement in low-altitude urban environments. To solve these problems under uncertainties, mathematical models and algorithms are needed. However, these models and algorithms are lacking in the emerging UAS operational problems, which requires considerable efforts to bridge these research gaps. Therefore, this thesis focuses on developing a Third-party Risk Assessment and Mitigation System (TRAMS) by leveraging mathematical optimization and machine learning techniques. TRAMS considers both ground risk and air risk with the objective of minimizing safety risks and improving the efficiency and resilience of UAS operations.

First, ground risk assessment and mitigation problems are solved by a novel UAV flight path optimization method that considers an integrated cost assessment model. The assessment model incorporates fatality risk, property damage risk, and noise impact, which is an extension of current third-party risk indicators at modeling and assessment levels. To solve the proposed optimization problem, a hybrid estimation of distribution algorithm (EDA) and CostA* (named as EDA-CostA*) algorithm is proposed, which provides global and local heuristic information for path searching in cost-based environments. Simulation results demonstrate the effectiveness and reliability of the developed optimization model in reducing risk costs.

Second, aiming at handling uncertainties in ground risk studies due to people's movement, a two-stage stochastic route optimization model is proposed to jointly make flight approval and execution decisions. In the first stage, the approval decision is made by checking the feasibility of flight missions based on the target level of safety and flight duration to eliminate unsafe flight plans. The route selection and departure time decisions are made in the second stage by considering stochastic risk conditions to further reduce the ground risk. Numerical studies demonstrate that our proposed stochastic route optimization model performs the best in fatality risk reduction compared to distance-based and conventional risk-based methods.

Third, air risk assessment and mitigation are conducted to prevent airborne flight conflicts. A decision-making framework is proposed to adaptively select the most suitable resolution strategy for different types of flight conflicts. The proposed framework is formulated as a double-layer optimization problem considering scheduling, speed adjustment, and rerouting strategies for conflict resolution. The framework's first layer is established to decide which strategy should be selected for what type of conflict. The second layer is developed as a mixed-integer nonlinear programming (MINLP) model to optimize the decision variables of the strategies selected by the first layer. The model is solved by a novel meta-heuristic stochastic fractal search (SFS) algorithm. Simulation results show that the proposed framework can resolve all conflicts with little cost of flight delays.

Finally, to cope with uncertainties (e.g., flight delay and trajectory deviation) in conflict resolution decisions, a chance-constrained UAM traffic flow management (UTFM) optimization model is proposed to generate robust conflict-free trajectories and reduce flight delays. The model includes a probabilistic constraint of separation requirement to cope with trajectory time deviations caused by environmental disturbances, such as wind. To solve the model, we convert the probabilistic constraint into a deterministic constraint with a risk-bounded separation guarantee. Numerical studies demonstrate the proposed model's effectiveness, robustness, and scalability. The proposed chance-constrained UTFM model forms an additional layer of safety assurance in the pre-tactical phase, which enables fast flight disruption recovery in the UTFM system under delay and trajectory uncertainties.

Chapter 1

1 Introduction

The rapid advancement of technologies in unmanned aircraft systems (UAS), commonly referred to as drones or air taxis, in and around urban airspace brings huge opportunities to unlock the potential of the sky [1], [2]. Because UAS operations not only generate considerable economic benefits but fast humanitarian assistance. For instance, companies such as Amazon, Meituan, and UPS are developing drone-based delivery services for small packages [3], [4]. While other companies such as Airbus, Ehang, and Joby Aviation are exploring urban air mobility (UAM) applications for passenger transport in urban environments [5], [6]. Drone delivery and urban air taxi services have great potential to reshape the traditional ways of people and cargo flows with faster speed and better experiences.

On the other hand, UAS operations also demonstrate the humanitarian benefits, such as medicine or vaccine transport during the Covid-19 pandemic [7], fire-fighting response and people rescue in high-rise buildings [6]. Which offers fast and effective assistance for people in need and prevents them from the risks of exposure to dangerous environments. What is more, UAS liberates us from labor-consuming tasks such as security patrol, disaster inspection, and traffic monitoring, which enables us to do some more creative work. People with recreational and educational purposes of using drones may also appreciate the UAS development, as it gives them unprecedented convenience and experiences to explore.

1.1 Motivation

Although the UAS industry has been developing fast in the last few decades, UAS operations in urban environments are still very limited due to the everlasting safety being hard to guarantee. For example, as shown in Fig. 1.1, a drone may crash and cause third-party risk to (i) third-party people on the ground that do not benefit from the drone operations, and (ii) third-party people on-board other aircraft are related to air risk [8]–[11]. Airborne collisions between drones yield air risk, and the falling of collided drones causes knock-on ground risk. These safety risks could result in severe casualties to third-party people and considerable property loss to urban infrastructures. What is more, there is very limited research that have been done on the risk analysis and prevention for emerging UAS operations, leaving the safety risk management of the UAS traffic system in its weak. This arouses our study on the development of a Third-party Risk Assessment and Mitigation System (TRAMS) by leveraging mathematical optimization and machine learning techniques.

The TRAMS integrates ground risk studies (Chapter 3, Chapter 4, and Chapter 5) and air risk studies (Chapter 6 and Chapter 7). The ground risk studies produce risk-minimizing routes for individual flight missions. Based on the generated risk-minimizing route, the air risk studies handle multiple flight missions to revolve flight conflicts and disruptions under uncertainties. Incorporated ground risk and air risk studies for third-party people, the TRAMS enables safe, resilient, and efficient UAS operations with flight recovery capability under uncertainties in urban environments.

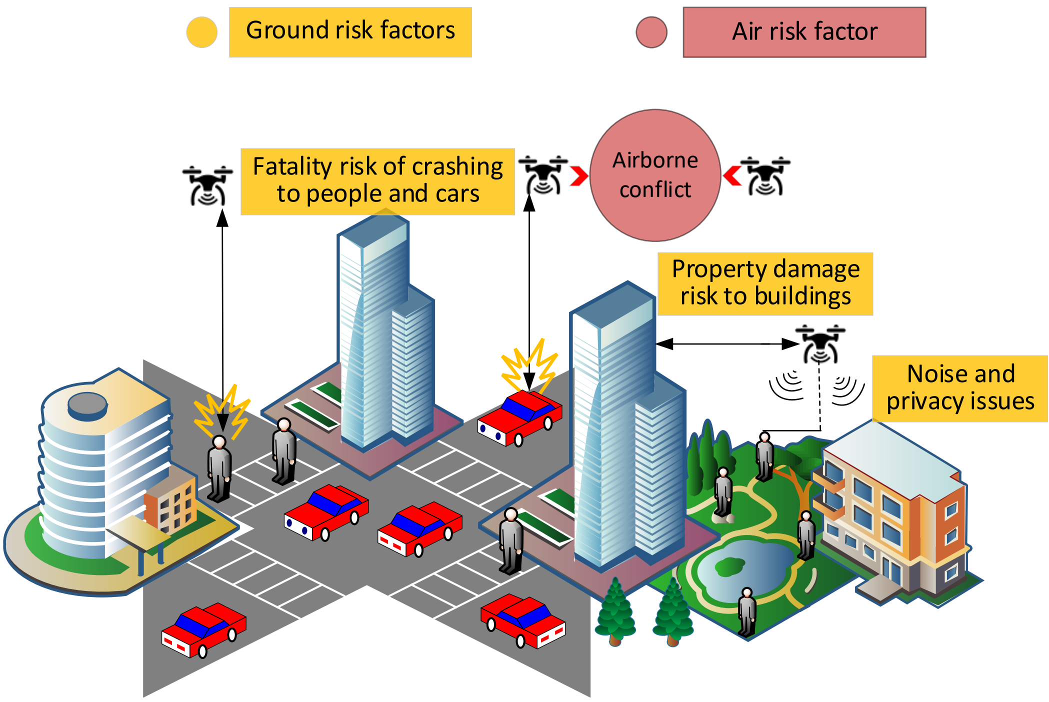


Fig. 1.1: Illustration of third-party ground risk and air risk caused by UAS operations in urban environments.

As mentioned above, TRAMS considers both third-party ground risk and air risk with the objectives of minimizing safety risks and improving the efficiency and resilience of UAS operations. Compared with non-risk-based UAS traffic management systems, the risk-based multi-UAV systems are increasingly important thanks to their huge benefits, such as:

- Safer UAS operations. The risk-based system takes a wide range of third-party risks into account, and these risks are identified, assessed, and used when conducting route planning and decision-making. This enables UAVs to avoid high-risk areas, such as locations with dense populations and vehicles (e.g., city square), critical infrastructures (e.g., nuclear powerplant), and security facilities (e.g., airport and police office).
- A more resilient multi-UAV system. By generating conflict-free strategic 4D routes, the potential conflicts are solved before flight. While the flight recovery in the pre-tactical phase makes the multi-UAV system more resilient to handle various uncertainties such as flight delays and trajectory deviation due to wind.
- More efficient management of airspace resources. By conducting risk-based route planning and decision-making for each flight plan, we can maximize the utilization

of limited urban airspace and create more supplies to meet the increasing demand of UAS operations.

- Better societal perception and acceptance. The societal acceptance of UAV operations in urban is one of the biggest barriers to developing UAS applications [12]. By conducting risk-based planning, third-party risks such as fatality risk, property damage risk, noise, and privacy impacts are evaluated and mitigated. That helps to win support from the public to implement UAS operations.

Above-discussed benefits greatly attract attention to UAS research and development. The safety risk is one of the major challenges for UAS traffic management in urban environments. Efficiency and resilience are also important aspects for the development of the UAS traffic system. These challenges have been given wide consideration by industry and academia. However, there are still quite a few significant research gaps that need to be filled to enable safe, resilient, and efficient UAS operations in urban environments.

(1) Safe operations require quantitative modelling and accurate assessment of third-party risks to conduct risk mitigations. What is more, risk-based route planning methods are not well defined and developed to solve risk-minimizing route planning problems effectively and robustly.

(2) Existing works treated ground risk assessment and mitigation statically and separately, which fails to provide support for time-dependent decision making. What is more, the studies on time-dependent route optimization under stochastic risk conditions have not been seen to jointly optimize flight approval and execution decisions for emerging UAS operations.

(3) Intelligent decision-making approaches for 4D trajectory optimization and flight conflict resolution are not well developed to adaptively solve different types of conflicts with tailored strategies.

(4) Models are lacking for disruption recovery of UAS traffic flows that are more vulnerable under uncertainties of individual flight missions and challenging weather conditions in low-altitude urban airspace.

The above-discussed research gaps motivate us to develop a third-party risk assessment and mitigation system (TRAMS) to achieve safe, resilient, and efficient UAS operations in urban low-altitude airspace.

1.2 Objective and Scope

The objective of this research is to study risk-based route planning and decision-making for safe UAS operations in urban environments. This thesis aims to investigate and answer the following four questions: (1) how to model and assess third-party risks (fatality, property damage, etc.) in urban environments? (2) how to formulate and solve the risk-based UAV route planning problem to minimize the safety risk in the planning phase. (3) how to adaptively resolve flight conflicts among multi-UAV routes and generate conflict-free 4D routes for UAS operations? (4) how to handle uncertainties effectively and resiliently in risk assessment and route optimization decisions.

To make the scope of this thesis clearer, the following assumptions are made:

- This thesis studies risk-based unmanned aircraft traffic management problems and will not investigate the behavior of individual UAVs at the robotic level with motion control decisions.
- The risk mitigation in this thesis is conducted from the perspective of air traffic management. Other mitigation approaches such as equipping parachutes for UAVs or using lighter and softer manufacturing materials are not considered in this work.
- The risk of a midair collision between a UAV and commercial aircraft is not considered in this work, as the aerodrome control zone is treated as a restricted area where the UAV is strictly not allowed to enter. The risks of UAVs intruding security-related bases and facilities are also out of the scope of this study.
- Unmanned aircraft refers to both quadrotor UAVs and fixed-wing UAVs. To cope with both types of UAVs, hovering is not made as an option in the problem formulation of this thesis, as fixed-wing UAVs are unable to hover at the speed of zero.
- This thesis provides fundamental developments of concepts of operations, mathematical models, and algorithms, which can be applied to UAS traffic management (UTM), urban air mobility (UAM), and advanced air mobility (AAM) for risk assessment and mitigation, flight trajectory, and traffic flow optimization.

1.3 Contributions of Thesis

This thesis studies third-party risk assessment and mitigation problems for safe, resilient, and efficient UAS operations in urban environments. The key research contributions of this thesis are highlighted and presented in Fig. 1.2.

First, we introduce a risk-based AirMatrix concept of operations (ConOps), which enables digitalized low-altitude urban airspace management by incorporating various data sources, such as third-party risk, flight plans, communication, navigation, and surveillance of unmanned aircraft. The risk-based AirMatrix ConOps provides both safety assurance and flight flexibility for unmanned aircraft system (UAS) traffic management.

Second, we propose a novel flight path optimization method that considers an integrated cost assessment model. The assessment model incorporates ground fatality risk, property damage risk, and noise impact, which is an extension of current third-party risk indicators at modeling and assessment levels. The developed method allows the quantitative evaluation and mitigation of ground risk by generating risk-minimizing 3D routes for each flight.

Third, we develop a two-stage stochastic route optimization model to jointly make flight approval and execution decisions to minimize the total ground risk of UAS operations. To the best of our knowledge, this is the first time that the risk-based route optimization problem has been resolved under stochastic risk conditions caused by population movement.

Fourth, we propose an adaptive decision-making framework for multi-UAV conflict resolution considering scheduling, speed adjustment, and rerouting strategies with investigations of their inner mechanisms. The adaptive framework contributes to the conflict resolution for air traffic control and management of emerging UAS operations in low-altitude urban airspace.

Fifth, we develop a chance-constrained unmanned air traffic flow management (UTFM) optimization model with fast disruption recovery to solve observed disrupted flights (reactive approach) and improve their resilience in response to future uncertainties (proactive approach). The UTFM model adds an additional safety assurance layer for the UAS traffic management system.

Lastly, we propose a third-party risk assessment and mitigation system (TRAMS) by integrating ground risk studies (Chapter 3, Chapter 4, and Chapter 5) and air risk studies (Chapter 6 and Chapter 7). The ground risk studies produce risk-minimizing routes for individual flight missions. Based on the generated risk-minimizing route, the air risk studies handle multiple flight missions to revolve flight conflicts and disruptions under uncertainties. Incorporated ground risk and air risk studies for third-party people, the TRAMS enables safe, resilient, and efficient UAS operations with flight recovery capability under uncertainties in urban environments.

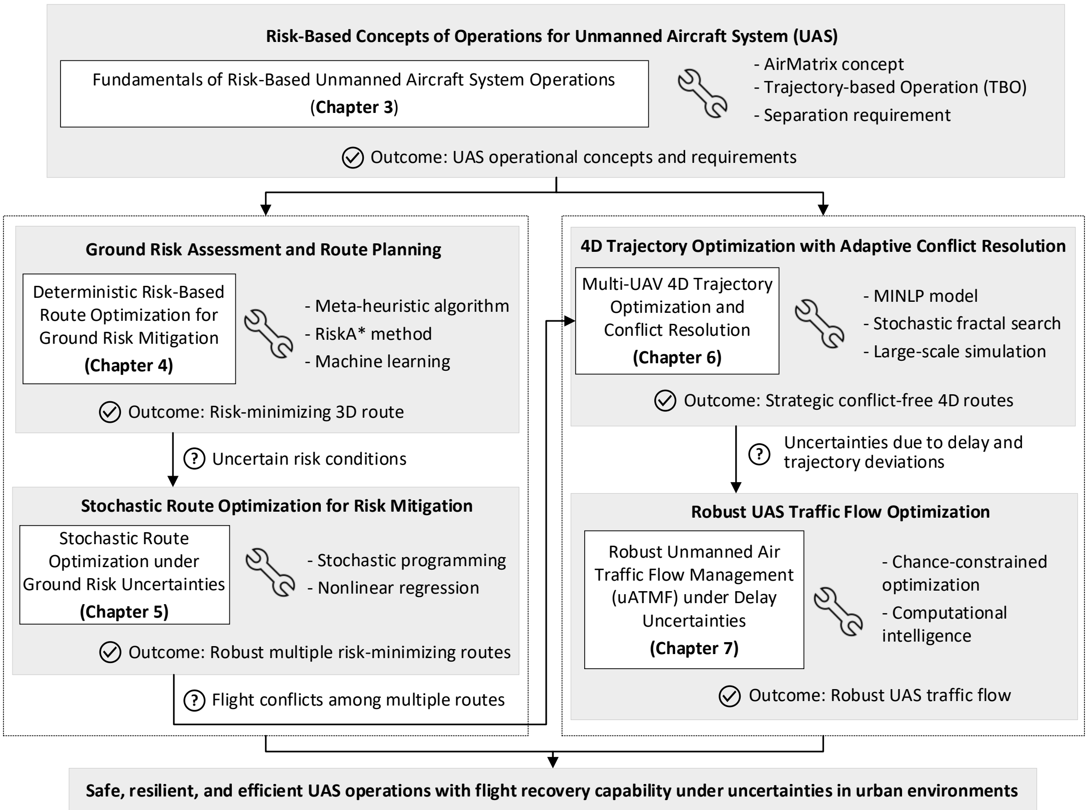


Fig. 1.2: Research framework and contributions of the thesis.

1.4 Outline of Thesis

The rest of the thesis is organized as follows.

Chapter 2 conducts an extensive literature review from perspectives of (i) third-party risk modelling and assessment, (ii) ground risk mitigation through risk-based route optimization, and (iii) air risk prevention by conflict detection and resolution. Key research gaps are identified and summarized at the end of this Chapter.

Chapter 3 provides the fundamental concepts of operations for risk-based airspace management and UAS traffic flow management under the AirMatrix framework.

Chapter 4 presents the modelling of a novel UAV flight path optimization method that considers an integrated cost assessment model. The model enables the quantitative assessment of ground risk for an urban area, and the generation of risk-minimizing routes for individual UAV flights.

Chapter 5 proposes a two-stage stochastic route optimization model to handle uncertainties in ground risk studies. We first study the risk-based route optimization problems under stochastic risk conditions caused by population movement. The proposed model allows the generation of robust multi-UAV risk-minimizing routes.

To solve the flight conflict among multiple flight routes, Chapter 6 builds a double-layer decision-making framework to adaptively select the best suitable resolution strategy for different types of flight conflicts, which subsequently generate conflict-free 4D flight trajectories.

Chapter 7 studies the safe and resilient UAS traffic flow problems and develops a chance-constrained UAM traffic flow management (UTFM) optimization model to cope with uncertainties (e.g., flight delay and trajectory deviation) in conflict resolution decisions.

Chapter 8 concludes the key research findings of the thesis and discusses the possible directions of future works based on our study.

Chapter 2

2 Literature Review

Route planning and decision-making are one of the most critical and active research topics in the air traffic management field for both manned and unmanned aircraft. In this Chapter, we conduct a comprehensive literature review on research topics of (i) third-party risk modelling and assessment, (ii) single UAV risk-based route planning, and (iii) multi-UAV route optimization and decision making for conflict detection and resolution. Key research gaps are also summarized at the end of this Chapter. Note that, as risk-based route planning and decision-making for UAS operations is an emerging area, more specific literature is also provided for each research problem in the following Chapters.

2.1 Third-Party Risk (TPR) Modelling and Assessment

Risk modelling and assessment for UAS operations have been widely studied [8]–[12]. UAS risk analysis is divided into two groups [13], which are the hazards and consequences. The hazards are factors that might be causing the failure of UAS operation, while the consequences are the potential damage the failed UAV may cause to human life or property. The hazards can be further divided into internal hazards and external ones. Internal hazards are the factors from UAV itself such as hardware issues and software issues. So, it can also be regarded as an airworthiness part. The UAS airworthiness certification has been studied [14] and a risk matrix concept was proposed to provide a structured classification of all risks. That helps regulatory bodies to make regulations based on the levels of risk. Other internal hazards were also investigated such as UAS propulsion system issues and battery issues [15], [16]. These are key hardware issues that can cause the total failure of UAS operations. Software issues are also important such as the flight control system [15] and related algorithms [17]. The malfunctions of the systems can result in a complete loss of control of UAS. On the other hand, external hazards are also critical, and the main factors are weather, signal condition, and obstacles. Convective weather (e.g., thundering, and heavy rain) can cause the failure of UAS operation [16], [17]. Bad signal conditions cause the loss of communication and control, thus resulting in the failure of UAS [18]. Navigation signal condition also matters, as the loss of navigation can cause great flight deviations [19], which greatly increases the collision probability of UAS with obstacles in urban dense building areas [20], [21].

Failed UAS may cause damage to human life and property, and this arouses the consequences of UAS risk analysis. the consequence is mainly classified into three groups: fatality risk, property damage risk, and other risks [13]. Fatality risk is regarded as the primary risk type because it is pertaining to human life [13], [17], [22]. Falling UAVs may hit people and vehicles on the ground, causing fatalities. This type of risk is highly correlated with population density and traffic density [10]. Property damage risk is another important issue. Falling UAVs may hit the building and cause damage [23], [24]. UAVs operating in low-altitude airspace also bring societal issues such as noise impact and privacy concerns to the public [12]. These issues are considered psychological risk costs of UAV operation to the public, which need to be mitigated in the mission planning phase.

Above we discussed the existing works on the hazards (internal and external) and consequences of UAS risk analysis. We further analyze the works on modelling and assessment of these risk factors. The authors proposed a framework to conduct the qualitative risk analysis for UAV operations based on international safety standards and quantitative analysis using Bayesian network modelling [17]. With the risk analysis, the risk-cost map for UAS operations was proposed to help better manage the safety concerns of UAS [25]. Factors contributing to the risk management of UAS operations were also studied by analyzing historical accident data and conducting exploratory factor analysis [26]. Based on these frameworks and concepts [13], risk modelling and assessment can be classified into ground risk, air risk, and other risks.

The above studies from the research community are more focused on specific risk types modelling and assessment. While regulatory reports provide a conceptual framework for risk analysis. Joint Authorities for Rulemaking of Unmanned Systems (JARUS) proposed a framework for risk analysis of drone operations [13], [27]. The proposed specific operational risk assessment (SORA) classifies risk types as air risk, ground risk, and other risks. The risk values are contributed by their probability of happening and the severity of consequences. And the SORA also gives a detailed process to conduct the risk assessment for UAS operation. The International Civil Aviation Organization (ICAO) proposed the general safety risk management framework for UAS operation, which gives a global view of the UAS risk assessment [28]. In the following year, the Federal Aviation Authority (FAA) also issued a report that studied unmanned aircraft systems risk management [29]. [28]

2.1.1 Ground Risk

As to ground risk modelling and analysis for people on the ground, a third-party casualty risk model was proposed based on a target level of safety approach with even tree analysis, to predict the casualty of UAS operations [8]. The study found that the UAS weight and population density are two key factors influencing ground casualty. Clothier et al [10] studied the risk modelling of UAS for people on the ground and proposed a barrier Bow Tie model for risk analysis and mitigation. The proposed model can assist regulators to determine airworthiness regulations for UAS. There are also some existing studies focused on impact probability and severity models to people and vehicles on the ground. Pioneer works studied the probability of fatalities and the fatality rates associated with a ground impact on pedestrians, and analysis results showed that the risk of fatality to humans is low in the condition of light UAVs operating in areas with low population density [23]. Follow-up works studied the impact severity of UAVs to people, and they subsequently proposed the weight threshold of falling UAV impact ground people based on the injury scale and criterion [30]. The authors [31] reviewed and presented the main mathematic models for conflict and collision probability estimation, which provide insights into the collision risk assessment of UAS.

The risk assessment model of UAV to ground vehicles was also established. The authors [32] defined the possible ground impact area of falling UAVs and developed the collision probability model, which helps in the identification of high-risk areas of the road network. The risk of UAS impact obstacles due to limited supporting infrastructure was also studied [24]. Based on the UAV impact probability and severity studies, the researcher proposed a risk-based approach for small UAV operations [33] and generated the probabilistic map using Monte Carlo simulation for more accurate ground impact risk analysis [34]. By using these proposed models and methods, some practical studies have been performed to model the third-party risk in urban environments [35], [36], and the level of risk for the UAV system was also proposed to identify critical areas and actions [15]. The analysis and modelling of these risks facilitate the generation of a risk-aware map, which can be used for risk mitigation by conducting airspace management and UAV route planning [22], [37].

2.1.2 Air Risk

Air risk is another branch of risk analysis, which consists of the airborne conflict risk between UAV and UAV, and UAV with crewed manned aircraft and other flying objects [38]. To prevent air risk, conflict detection and resolution studies have been studied with various types of strategies. Scheduling is an effective way to tackle pre-flight conflicts. By adjusting the departure time, potential conflicts or congestion can be solved [39]. Compared with commercial aircraft, the scheduling of UAS operations has more challenges, as the UAS traffic network is more complex in low-altitude urban airspace. Another conflict resolution strategy is to adjust the flight speed, which subsequently adjusts the ETA for the succeeding waypoints of a flight [40], [41]. To overcome the uncertainty impact on the pre-planned 4D routes, one study [42] proposed a dynamic adjustment method to adjust the airspeed in response to the leads or lags of flight, which increased adherence to the estimated time of arrival at each waypoint in the pre-planned route. Rerouting is another effective way to tackle flight conflict. By adjusting the 3D geometric trajectory, potential conflicts can be resolved in the spatial dimension. Compared with the scheduling strategy, the rerouting strategy changes the flight trajectory to avoid conflicts, convective weather, and other risky areas [43], [44].

UAVs may also hit the critical components (e.g., engine and windshield) of manned aircraft and cause catastrophic accidents [45], especially in the vicinity of the airport. Studies investigated the collision risk of UAS posed on commercial manned aircraft in aerodrome areas [46], [47], and Monte-Carlo simulation was used to predict the collision probability within terminal airspace [46], [47]. And the authors developed a 3D Monte-Carlo UAS positional distribution model, based on the flight dynamics of the UAS, to help assess the risk posed by the UAS to aircraft operating inside the aerodrome. Follow-up works studied how to minimize the aerodrome alert zone, and this study helps to release more airspace for UAS operations [22], [48], [49]. The authors investigated the trajectory data of manned aircraft in terminal airspace. By clustering the trajectory data, they found the main feathers of the traffic flow and the unutilized volume of terminal airspace. The identified unutilized airspace could potentially be used for UAS operations. Air risk analysis faces several other challenges such as low-altitude urban airspace management [50]–[52], Airlink network design [53]–[55], and management for non-cooperative flying objects [56], [57]. Although these challenges are not in the scope of this work, there are potential research topics to explore.

2.1.3 Other Risks

The last branch for UAS risk analysis is other risks. UAS operating in low-altitude urban airspace brings societal issues such as noise impact and privacy concerns to the public [12]. These issues are of great concern and considered psychological risks of UAV operation to the public, which has the same priority to be mitigated before UAS operations can be largely implemented in urban areas. A study reported that the public is not well-accepted drone operations, as they see the drone operation as a risky activity that may hit them or directly interfere with their privacy [58]. And the authors recommended that the risk analysis and mitigation of drone operation in urban environments is critical before we can proceed to commercial and hobby applications. The follow-up work studied the public acceptance of drone applications with two factors that fears and concerns, and potential benefits can be obtained [12]. The author concluded that the public would say "yes" unless some of their key concerns can be solved such as fears and privacy issues. In another direction, the authors studied the effects of UAS noise on people [59]. They measured the reported loudness, annoyance, and pleasantness from the participants, and results suggested that drone flight paths along the road traffic can significantly help to mitigate drone noise issues, as the noise made by road traffic itself is well louder than the noise drone produces. To facilitate drone path planning for mitigating noise, the researcher investigated acoustic mapping for drone operations [60]. The generated noise exposure map identifies high-noise areas where the drone path should avoid. Another study [61] proposed the air-taxi noise emission model and found that the number of UAM vehicles is the evident factor to predict noise emission for air-taxi operations in urban environments. The similar reasoning also applies to safety risks posed to people on the ground [9], [11], [62]. As we can see, the other risks mainly contain noise and privacy issues from the public. The public is more concerned about the risk and privacy issues of drone operations.

In summary, the UAS risk analysis involves the hazards and consequences of drone failure and contains ground risks, air risks, and other risks. Among all the risk hazards, the weather and signal conditions attracted more research attention. For the consequences of drone operation, more studies are on fatality risks of people on the ground, while the public shows more concerns about noise and privacy issues. They can only accept extensive operations before the risk issues can be mitigated. However, fewer studies are on third-party risk assessment and mitigation. Noise issues are even less considered as an impact risk cost to the public, which needs to be quantitatively assessed and mitigated when UAVs operate in urban low-altitude airspace.

2.2 TPR-Based Single UAV Route Planning

As one of the key enablers for autonomous UAS operation, route planning problems have been extensively investigated. The route planning is to generate a feasible, optimal, sequence of waypoints while avoiding collisions for UAV moving. The 4D route planning is distinct from its 3D counterpart because of the involvement of time. For the optimization of the 4D route, there are two types of variables in discrete environments, explained in Fig. 2.1. One is the spatial variable, which consists of the 3D coordinates of each UAV waypoint. The other is temporal variables, and they are estimated time of departure (ETD), estimated time of arrival (ETA) of destination, and ETA of passing each waypoint. Speed is also considered a time-related variable, as the adjust of speed changes the ETA for the following waypoints. We discuss the variables in a discrete manner, and they are the ones to be planned and optimized in 4D route optimization problems.

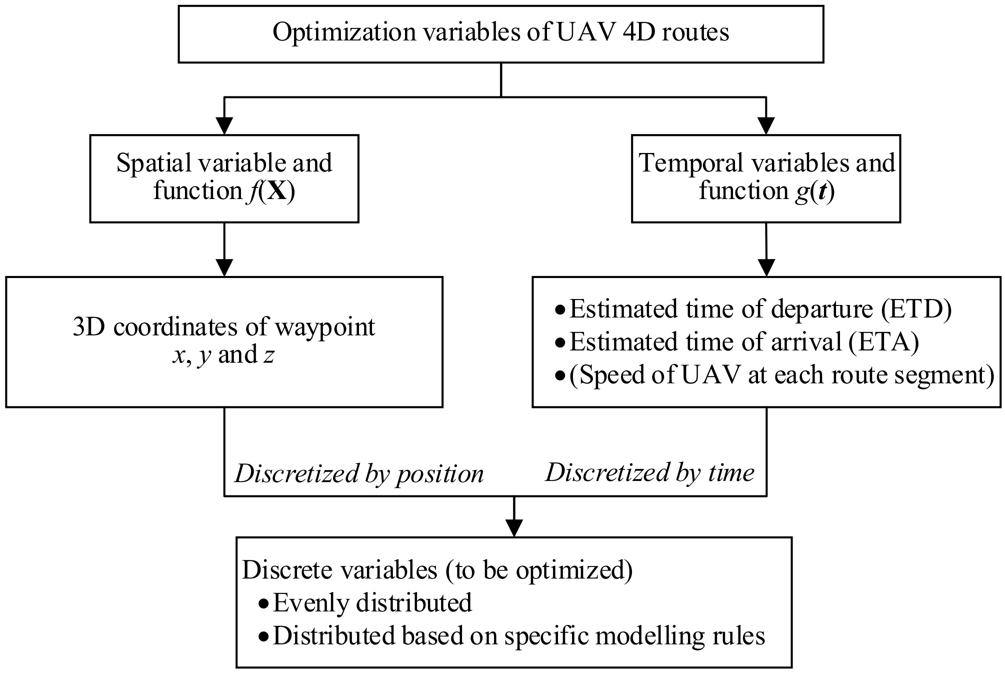


Fig. 2.1: Decision variables in the 4D route optimization problem.

The objectives of route planning can be classified into (i) minimizing flight distance (or time) and (ii) minimizing safety risk of the flight route, which can be referred to as distance-based and risk-based planning, respectively. We conduct a comprehensive literature review to discuss the main methods and algorithms used for both planning types.

2.2.1 Distance-Based Route Planning Methods

For distance-based route planning, the methods mainly include graph-based decomposition, mathematical modelling, potential field based, and sample-based methods.

A. Graph-based method

Graph-based methods are extensively used for single UAV path and route planning [63]–[65]. The idea of graph-based methods is to construct a discrete directed or undirected graph where the vertex represents a collision-free node, and the edge presents a unique collision-free path or arc. The planned path or route is a series of waypoints that connects the start point and end point. On the node and edge, there is always a weight representing the cost of being chosen to travel. With this graph structure, exact methods such as the Dijkstra algorithm, or swarm intelligent algorithms such as genetic algorithm, ant colony algorithm, and A* algorithm are employed to find a feasible, even optimal, solution always with objectives to minimize flight distance and flight time or to maximize utilization of UAV. Authors [63] modelled the travel salesperson (TSP) to maximize the collected information using drones, and a modified genetic algorithm was used to find the solution while avoiding the forbidden areas. Works were also conducted on the comparison of 3D path planning and 4D trajectory planning [64]. The 4D one considered the time of passing each path point and it performed better than the 3D one in terms of predictability and safety, whereas the computation time is greater, especially in complex environments. To deal with this issue, other researchers [65] proposed the evolutionary optimization algorithm incorporated with t-distribution to reduce the computational complexity.

B. Mathematical programming method

Mathematical modelling is another widely used method. UAV path and route planning problems are always modelled as linear programming (LP), integer programming (IP), mixed integer linear programming (MILP), and mixed integer nonlinear programming (MINLP) problems for different missions with various optimization objectives and constraints [66]–[69]. The integer programming model was developed to solve the routing problem for UAVs with considerations of human factors and workload, and the heuristic algorithm was devised to solve the IP problem [66]. To improve the solution quality, the authors proposed a new evolutional method to decompose the path planning with better waypoints being exploited [67]. By considering the time dimension, some researchers studied the time-dependent TSP problem and constructed the integer programming model to minimize operational costs [68]. And a heuristic Branch and Cut algorithm was proposed to solve the modelled problem to optimality. Apart from offline planning, some authors investigated the dynamic path planning problem [69]. They developed a multi-objective path planning framework considering the safety index to minimize the risk of the planned path. Results show that the proposed model can be conducted online with acceptable latency to avoid unexpected obstacles. Robustness and reliability are two important indicators for route planning methods and algorithms. From a mathematical theory point of view, authors conducted several great works [70]–[72]. Dimitri investigated the robust shortest path planning problem and proposed a minimax formulation. Dynamic programming was employed with semi contractive model. Another work studied the reliable path planning problem for drone delivery and a stochastic time-dependent network was proposed to enlarge the delivery range [72]. The above-discussed methods model the path/route planning problems as proper optimization problems. As we can choose any resealable objectives and constraints, that empower the mathematical model with great applicability for various real-world problems.

C. Sampling-based method

The sampling-based method has its unique advantages of implicitly exploring the obstacles and generating collision-free routes. The fast processing speed makes it popular in dealing with online route planning and conflict resolution problems [73]–[76]. Two main sampling-based methods are the probabilistic roadmap method (PRM) [77] and the rapidly exploring random tree (RRT) method [73], [74]. One thing they have in common is that their fundamental scheme is sampling. The PRM has only a one-round sampling process while RRT uses the incremental sampling process. Once the start point and end point are determined, the PRM will generate a whole sampling road map for given environments. The RRT starts from the origin point (or destination point, backward RRT) and generates the collision-free configuration. The nearest point to the current one will be selected as the next point to move and the process evolves till reaching the destination. This character of RRT makes it capable of dealing with online fast route planning conflict resolution to avoid dynamic obstacles [75]. Over the years, RRT methods have been extensively used in UAS online path planning [73], route optimization [74], real-time conflict resolution [75] and motion planning [76] in various fields and environments. The application of these methods in UAS route optimization and real-time decision-making would be another promising branch.

D. Potential field based method

Potential field (PF) based methods are another widely used continuous approaches to solve route planning, conflict resolution, and network optimization problems [52], [78]–[80]. The idea of potential field based methods is that the destination has an attractive potential field to appeal agent moving toward the target while obstacles have a repulsive potential field to repel the agent from collisions. Potential field methods have first been extensively used in robotics in applications of real-time obstacle avoidance [81] and navigation [82] for mobile robots. In recent years, they were also introduced in transportation fields such as UAV route planning [80], obstacle avoidance [79] and target-following [78], [79]. These studies show that the PF methods have a good performance in point-to-point (origin to destination or object to obstacle) potential field modeling and planning with fast computational speed. These findings will benefit large-scale UAS route optimization and conflict resolution.

2.2.2 Risk-Based Route Planning Methods

Risk-based methods aroused in recent years due to the critical need for risk mitigation for UAS operating in urban environments. This method contains the risk modelling and assessment part to generate the risk-based map for route planning. Algorithms employed to cope with risk-based route planning are various based on the different fitness functions developed [22], [83]–[86]. Traditional methods have various optimization objectives such as minimizing travel distance and cost and maximizing flight duration. However, these methods are always considered obstacle avoidance but ignore risks underneath the UAV flying route. Extended from conventional distance or cost-based path planning problems, the risk-based methods relied on risk maps for path planning [22], [83]. Various methods and algorithms were developed to generate the risk map and to cope with the risk-based path planning problems. The A*-based algorithms, for instance, were developed together with Dubins Curves for risk-based path planning and smoothing [83]. In the follow-up study, the authors [85] developed a RiskA* algorithm to minimize the risk of the produced path. Genetic algorithm and Dijkstra methods are also popular in addressing this problem and have been chosen as benchmarks to compare with A*-based path planning algorithm [37], [87]. Other methods such as the Markov decision process used a hierarchical method to maximize efficiency and minimize risks [84]. A rapidly exploring random tree (RRT) was proposed to minimize the third-party risk of UAV takeoff trajectories [88]. And Tabu search algorithm was employed to optimize the UAV route to minimize the cost of damaged cargo [89]. What is more, authors [90] also developed a risk-aware graph search algorithm to select paths that have a high probability to yield low risk. On the other hand, the UAV operational environments have also been covered from factory-like areas [84] to inhabited areas [91] and to urban environments [92]. In different areas, the risk types are various. For instance, in factory areas, the main risk sources are critical infrastructure and property damage. While in inhabited areas and urban areas, population and vehicle density, high-rise buildings are more sensitive for risk-based path planning.

In summary, for single UAV route planning, the graph-based method and mathematical programming are two main branches. The sample-based method performs well in exploring the implicit obstacles and online conflict resolution while potential field-based models work well in continuous route generation and real-time conflict resolution problems. As multi-agent systems and reinforcement learning methods mainly solve the interaction and collaboration problems among multiple agents, their uses for single UAV route planning are still lacking. Furthermore, risk-based route planning methods have been studied less for emerging UAS operations in urban environments. Where the distribution of risk cost in each grid cell is unpredictable and uneven, making the risk-based route search a difficult problem.

2.3 TPR-Based Multi UAVs Trajectory Optimization and Decision Making

We have discussed the single UAV route planning problems, and various methods and models have been presented and surveyed for optimizing the single route. In real-world operations, multiple UAVs are operating in airspace. It is necessary to study cooperative planning of multiple UAV flights that consider the accumulated safety risk posed by many UAV flights in an urban area. There are two aspects to investigating cooperative planning and decision-making. One is to conduct multi-UAV trajectory optimization to generate feasible or even optimal flight timetables and routes. The other is to perform cooperative decision-making for conflict detection and resolution to reduce air risk. We give a literature review on these two aspects regarding the overall cooperative planning and decision-making strategies.

2.3.1 Multi-UAV Trajectory Optimization

Cooperative planning for multi-UAV mainly includes task allocation, scheduling, and route planning problems. Cooperative UAVs belong to the same group and can share information for achieving the same goal. Distributed control and decision-making methods can be used for cooperative operations. Noncooperative operations have different tasks and may not be able to communicate with each other. Therefore, concertized control and decision-making methods can be used in this kind of operation. Due to the limited airspace capacity and resources, it may not be able to support all

UAVs operating at the same time. Scheduling is to arrange, control and optimize airspace and timing resources so that the task requirement of multiple UAVs can be satisfied. As single UAV route planning and optimization is part of multi-UAV planning, the fundamental methods used for single planning are also applicable in multi-UAV planning. There are still some new features for multi-UAV route planning and optimization, as the number of UAS involved increases and applications become more complicated. The widely used methods for multi-UAV cooperative planning can be concluded as follows.

Graph-based methods are also extensively used for multi-UAVs route planning. Same as the single UAV planning method, the graph-based method is to construct a graph with collision-free vertices (nodes) and edges (links). Route planning is to find a special case of minimum cost flow based on the constructed graph. Authors studied the four-dimensional (4D) route planning problems for multi-UAVs in various environments [93], [94]. They modelled the multi-objective problems with the motion control to produce the collision-free route, while the online replanning functions were also considered to deal with unexpected obstacles. Particle swarm-based and hybrid optimization algorithms were proposed to solve their proposed models. Follow-up studies compared the performances between the genetic algorithm (GA) and PSO for online route optimization [95]. Various scenarios were proposed to compare the two types of algorithms, and statistical studies were also made to see the significance of the results. Their results showed that GA performs better than PSO in terms of route quality.

Mathematical programming also played a key role in solving the UAS route optimization problems. It has the advantage of modeling both discrete and continuous problems with various objectives and constraints. The authors [96] developed a path-planning model using partially observable Markov decision processes (POMDPs) to track multiple ground targets. Delivery missions using multi-UAVs were also modelled as MILP problems and heuristic simulated annealing (SA) has been improved to solve the proposed model [97]. Some authors paid attention to heterogeneous vehicles to deal with delivery and exploration missions [98]–[100]. To solve their proposed models, the authors developed hybrid GA to improve the quality of the route, a modern evolutionary approach to minimize total flight distance, and an adaptive large neighborhood search metaheuristic to solve large-scale problems to optimality. Based on that, researchers proposed a two-layer mission planning model to improve the survivability of the drone while ensuring the quality of solutions [101]. The flying sidekicks TSP problem was modelled to solve delivery missions using multiple drones [102], and MILP and probabilistic planning were constructed to better ensure the constraint satisfaction [103]. Different from single UAV planning, the mathematical methods used for multi-UAV problems consider the scheduling for both cooperative and noncooperative UAS operations. That greatly increases the computational complexity compared with the pure route planning model. To reduce computational time, several heuristic algorithms have been designed to find optimal solutions.

2.3.2 Decision-Making for Conflict Detection and Resolution

A. Pre-Flight Decision Making for Generating Conflict-Free 4D Routes

Uncertainties may occur before or during UAV flights such as malfunction of UA, instruction from air traffic control agent, demand changes from the customer side, etc. Some of the issues are significant and cannot be solved by local planning. Global mission changes are required to replan a feasible, even optimal, flight plan. Global change presents the four-dimensional pre-planned route is completely changed, which involves operational time and a 3D route. There are extensive studies [104]–[109] worked on the global mission change problems in various fields such as manned and unmanned air traffic management, autonomous cars, intelligent transportation system, etc.

In the civil aviation field, authors studied the scheduling and mission replanning problems for air traffic management. The authors modelled the 0-1 integer programming to optimize the runway throughput and schedule and reschedule the aircraft with consideration of safe separation. As airline flight operation is a sequential process, rescheduling or rerouting causes downstream impacts on following flights [105]. The authors studied the impacts of two main rerouting strategies called distance saving and distance increasing reroutes. They found that the former performs better and is more likely to be conducted in one of the major air transportation markets. Based on that work, other authors investigated the traffic flow dynamic model for scheduling and routing. The developed an ILP model to maximize the capacities of the airport and en route airspace [106]. A recent study [110] developed a pre-departure flight planning method to generate 4D conflict-free trajectories by assigning flights on different flight levels given the OD and desired departure time of each flight. This study has been demonstrated by simulations with a real-world city environment for urban air mobility operations.[108]

The scheduling problems are complex in terms of computation, and the feasible region could be nonconvex. To solve the nonconvex optimization problems for drone scheduling and path planning, the authors investigated an approach to transform a nonconvex problem into two convex subproblems without sacrificing the optimality of the solution [111]. They also proposed an exact algorithm to solve the transformed convex problems. What is more, some authors paid attention to modeling the more robust methods and algorithms for task assignment and path planning problems [109], [112]. They introduced iterative strategies to improve the performance of the task assignment algorithm and their methods improved the stability of proposed algorithms while reducing the computational complexity. To consider the mission priority in task assignment and scheduling problems, some studies proposed priority-based traffic scheduling models [107], [113]. In these studies, resources were provided to meet the demand for high-priority missions. Several applications in bus priority control and communication channel allocation were investigated to minimize the total passenger waiting time and to reduce the number of vehicles used.

Above mentioned works studied the pre-flight scheduling and mission planning problems using various models for several applications. However, there are still several research gaps that require more effort to meet: (1) most of the existing works conducted the scheduling and routing separately. They first planned the route and then for time; (2) a large majority of the existing scheduling studies focused on either OD mission or coverage mission, leaving the utilization of airspace resources in an unsafe and inefficient way.

B. Real-Time CD&R to Handle Flight Uncertainties

Real-time conflict detection and resolution is a major player in solving flight uncertainties in dynamic environments during the operation. On one hand, local change for conflict resolution is much more affordable compared with global mission change, as the latter causes significant downstream impacts for follow-up flights [105]. On another, most of these unexpected obstacles only interfere with a small segment of the pre-planned route, so UAV only needs to adjust the influenced part of the route. For instance, birds or other UAVs may interfere with a pre-planned route, but that interference only lasts for a few seconds as the route is cleared when these obstacles pass. To this end, local conflict resolution has attracted extensive attention in recent years. Studies are mainly on UAV motion control and 4D trajectory (4DT) adjustment to avoid collisions. As this thesis studies the problems in traffic planning and management sides, we do not focus on motion planning but on 4D trajectory adjustment for conflict resolution.

The four-dimensional trajectory includes 3D spatial dimension and time dimension. 4DT changes, therefore, are about the adjustment for the 3D path and the time of passing each path point [114]–[117]. The most frequently used way is to adjust the departure time of a flight so as to avoid detected potential collision on its trajectory [118]. The authors constructed an optimization model to conduct the routing and scheduling for large-scale UAS operations in urban environments. They first planned the 3D paths for all operations and then conducted the scheduling to arrange a feasible, aiming for optimal, departure time. Once pre-planned 4DT was determined, the potential collision can be detected using time separation. For instance, two consecutive flights must have more than 30 seconds of separation to pass the same waypoint. A penalty was incorporated into the optimization model for potential collision, which was reflected on the departure time of corresponded flight. Some other works focused on the time uncertainty and time of passing waypoint [116], [119]. Uncertainties during flights make pre-planned trajectories fail. To capture the flight time uncertainties and minimize the influence on the original 4DT, the authors proposed chance-constrained programming to solve that issue. While pilot attention is another source of uncertainty [116]. A time geography concept was proposed, and a time-space prism scheme was developed to capture the pilot's attention uncertainty during flight. The proposed method performed well in dealing with conflict detection and resolution.

Another branch of existing works studied the 3D spatial path adjustment for conflict resolution. Local rerouting is one of the primary players [43], [119], [120]. Researchers use mathematical programming models to construct timely rerouting, and effective algorithms such as Branch-and Cut have been proposed to solve the problems. Other studies investigated the routing problems when aircraft encounter convective weather during flight. What is more, the prohibited, restricted, and dangerous (PRD) areas are other sources of threats that are required to be avoided for aviation operations [43], [52]. In these studies, some researchers modelled the airspace in a discrete manner and developed a cellular automation algorithm to solve the rerouting problem with the objectives of minimizing collision risk and travel distance. Other researchers, in contrast, use continuous methods such as potential field-based models to conduct the rerouting and network optimization.

Besides, some authors used line-of-sight vectors and relative velocity vectors to perform conflict resolution for cooperative UAVs and formation missions [121]. While a new form of potential field based method [122], called collision-avoidance vector field was proposed to avoid both static and dynamic obstacles in complex environments for UAV operations. The proposed vector field method considered UAV kinematics and approximate speed modulation, which enables smooth and collision-free paths for multi-UAVs operations. B-spine curves have also been used to connect each waypoint and produced a continuous collision-free route [123]. As mentioned in previous literature, the RRT method conducts a single-query scheme. It builds a tree inclemently from the origin point to the destination point by sampling and selecting the nearest collision-free configurations at a fast speed. Based on that, RRT-based methods (DRRT, RRT*, etc.) have been extensively used for UAV conflict resolution, especially for online ones.

Reinforcement learning (RL) is another emerging approach to solve real-time conflict resolution problems. It has been used for solving sequential decision-making problems such as UAS route planning, resource allocation, and autonomous navigation [124]–[129]. The action policy is a key enabler of RL to plan the route from one status to another and to reach a desired goal. By using different forms of Markov decision models, the RL can cope with both discrete and continuous problems. Given an RL planner, in each status, the agent makes a pre-defined action to maximize the reward value function based on the current observation. It performs as an approximate dynamic programming to estimate the gradient vectors in the optimization process. And it has great advantages that does not rely on historical data but on observations of surrounding environments. Which enables the RL method to solve complex tasks in unknown environments even with partial observability. One of the works studied the resource balancing and allocation problem in a complex logistics network [124]. The standard multi-agent reinforcement learning (MARL) was improved with the novel cooperative mechanism for state and reward design, which efficiently and effectively improves transportation performance.

Other studies investigated the UAV autonomous navigation problem in complex environments with partially observable information [125], and MARL has also been used for heterogeneous vehicle route planning and scheduling for monitoring and delivery missions [130]. In recent years, deep learning has been incorporated with MARL and formed new approaches named multiagent deep RL (MADRL), which enables us to solve more complex tasks with larger scales [131], [132]. Although the MARL and MADRL approaches have unique advantages compared with traditional graph-based and mathematical programming methods, there are also several challenges such as the robustness, explainable ability, and applicability to solve real-world problems [128], [133].

2.4 Research Gaps

Existing works have studied the route planning and decision-making problems for UAS operations. However, there are a number of critical research gaps that need to be filled to cope with the unique challenges posed by the expected high-density UAS operations in complex and uncertain urban low-altitude airspace. Key research gaps are summarized as follows.

(1) Safe operations require quantitative modelling and assessment of third-party risks to conduct risk mitigations. What is more, risk-based route planning methods are not well defined and developed to solve risk-minimizing route planning problems effectively and robustly.

(2) Existing works treated the ground risk assessment and mitigation in a static and separate manner, which fails to provide support for time-dependent decision making. What is more, the studies on time-dependent route optimization under stochastic risk conditions have not been seen to jointly optimize flight approval and execution decisions for emerging UAS operations.

(3) Intelligent decision-making approaches for 4D trajectory optimization and flight conflict resolution are still lacking to adaptively solve different types of conflicts with tailored strategies.

(4) Models are lacking for disruption recovery of UAS traffic flows that are more vulnerable under uncertainties of weather conditions and flight delays in low-altitude urban environments.

The above-discussed research gaps motivate us to develop a third-party risk assessment and mitigation system (TRAMS) by leveraging mathematical optimization and machine learning techniques to achieve safe, resilient, and efficient UAS operations in urban low-altitude airspace.

Chapter 3

3 Concepts of Risk-Based Unmanned Aircraft System Operations

This Chapter introduces the basic terminologies and concepts such as the AirMatrix concept and risk-based airspace, which are used through this thesis. The third-party risks in urban environments are also investigated with detailed modelling and assessment, which provides risk mapping information for routing planning and decision-making.

3.1 Risk-Based Airspace with AirMatrix Concept

With the development of unmanned aircraft systems (UAS), commercial applications are rapid and booming. The Federal Aviation Administration (FAA) has estimated that the number of registered drones has reached 1 million and over 80% of them are under recreational licenses [134]. The rapidly increasing number of drones is a great challenge to regulating bodies such as the FAA and the Civil Aviation Authority of Singapore (CAAS). Large-scale UAV operations in low altitude and complex airspace may compromise the safety (e.g., midair collision) and efficiency (congestion and delay) of urban air transport. The need to manage and optimize UAS operations both safely and efficiently is crucial and urgent. To deal with these issues, numerous research has been conducted in areas such as detect and avoid, path planning and mission scheduling, etc. However, the problems of how to integrate UAS operations into current airspace and what kind of operational rules these operations should follow are still left unsolved.

Studies on the concept of operation for UTM systems are emerging areas. Pioneering works have been conducted to explore UTM ConOps [135], [136] not only from the whole framework but detailed elements which consist of airspace management, operational rules, flight scenarios, etc. Urban Air Mobility (UAM), as a new branch, has developed fast in recent years based on expected fast growth in passenger air-taxi and air-cargo transportation. To enable this kind of operation in the current airspace and air traffic management system, a concept of operation has been proposed [137] in terms of airspace management, operational rules, traffic management regulations, essential infrastructures, etc.

The first UTM conceptual framework was proposed by NASA. After that, studies have been conducted to keep the evolution of the UTM framework moving till a refined and comprehensive UTM ConOps was issued in 2018. The ConOps for airspace management was well presented and discussed in the operational concept part with objectives to illustrate key conceptual and operational elements regarding UTM operations. With the evolution and development of the UTM system, the UTM ConOps v2.0 has been released. One of the major updates of this version is taking more complex operations into account, including operations across both uncontrolled and controlled airspace.

The concept of U-space as European drone outlook studies was proposed [138], evolved [139], and released [1] by the SESAR team focusing on airspace management and safety aspects to support UTM operations from the user's point of view. The airspace rules and procedures for UTM operation have been well investigated and definitions for different airspace volumes were proposed. Flight rules, risk assessment, conflict management, etc. were also discussed. Another key component of the U-space ConOps is safety and social aspects. The ground risk, airport-related risk, and air risk have been analyzed, while social privacy and perception of drone operations in urban environments were also described. Contingency plans as the elements in this part were studied, and several new frameworks and procedures have been proposed for safety management. Further, the concept of operations for specific operation risk assessment (SORA) [13], [27] and risk cost map [22], [25], [85] for drone operations were studied considering operational risks and navigational deviations [19], [140].

In another group of state-of-the-art studies [55], [141], [142], researchers from Air Traffic Management Research Institute (ATRMI) at Nanyang Technological University combined and produced a ConOps with a labyrinth of unilateral digital lanes to support high-speed UAS operations. Based on that, a modular UTM system has been proposed and the core modules of the framework are urban airspace management, flight management, and risk management. Then, a preliminary ConOps was proposed for urban airspace management, which is the AirMatrix, a dynamic and scalable urban traffic management framework. In the framework[143], UAS route planning is an important component of UAV operations planning. Proper flight management and airspace management can significantly reduce the risk of UAS mid-air collision and improve the total utilization of urban airspace more UAS operations can be carried out simultaneously within specific airspace [14]. The dynamic AirMatrix traffic network and the risk-aware network have been proposed and discussed to provide safer and more efficient airspace and traffic management.

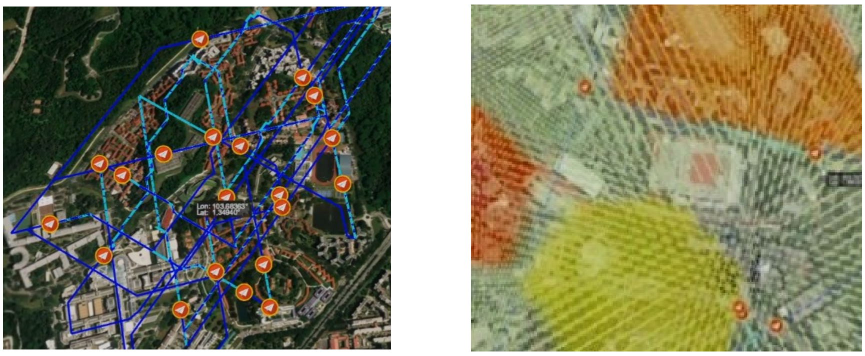
Existing UTM frameworks and ConOps are more focused on the big picture to describe the essential conceptual and operational elements and requirements regarding UTM operations, but fewer details were given for implementation purposes. What is more, existing ConOps considers airspace configuration and operational rules only from a few points of view, there is still a lot of room for improvement. In this regard, our work concentrates on the ConOps of UAS airspace configuration and operational rules, investigating the AirMatrix network (Fig. 3.1(a)) combined with different resolutions (Fig. 3.1(b)) and corridor (Fig. 3.1(c)). Free-flight operation and trajectory-based operation (Fig. 3.1(d)) are studied and compared through simulation studies. An airspace risk-aware map (Fig. 3.1(e)) and risk assessment procedure are proposed. The ConOps of contingency management regarding airspace management is also being studied and discussed in this work.

In this section, the AirMatrix ConOps is discussed with UAV movement rules in the proposed airspace network. To balance the airspace complexity and operational flexibility, the ConOps of different resolution AirMatrix is proposed. Further, to connect the different resolutions and segregated airspace areas, the AirMatrix corridor is introduced with detailed illustrations.

3.1.1 AirMatrix Airspace Concepts

The AirMatrix airspace concept describes an airspace structure (Fig. 3.2(a)), which provides discrete and standardized units to manage urban airspace, and the airspace is divided into uniform air blocks. This configuration allows quantitative analysis for a vast range of subjects. For example, the air blocks can encode information on command-and-control signal strength, population density underneath, etc. Such information can be served as metrics for safe flight planning. Therefore, the setting and properties of the AirMatrix can be dynamically adjusted according to the change in environments.

(a). AirMatrix (b). Different resolutions (c). AirMatrix corridor



(d). Trajectory-based operation (e). Risk-aware airspace

Fig. 3.1: Main components of airspace configuration and operational rule.

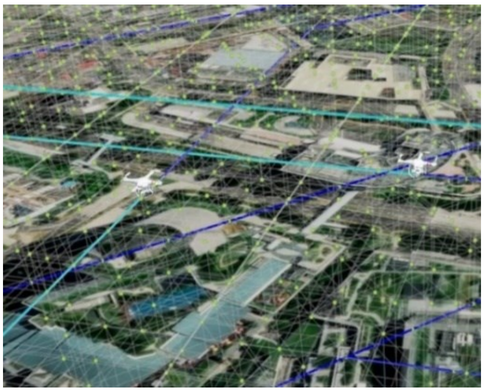
To make the statement clearer, terms used in this Chapter are defined below.

- Movement freedom: number of possible waypoints UAV can choose as the next waypoint to fly.
- Flight flexibility: number of air blocks in certain airspace.
- Traffic density: number of UAVs per unit airspace and per unit time.

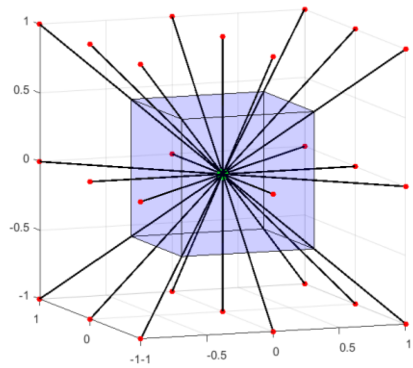
- Route crossing point: more than two air routes intersect with at a same altitude level.

- Airspace complexity: there are four contributors to airspace complexity in this Chapter. They are movement freedom, flight flexibility, traffic density, and crossing route points. In a given airspace, with the increase of all these contributors, the airspace complexity will rise accordingly.

Aerial vehicles (AVs) operating in such an airspace structure will follow the operational rules. One rule is how AVs move from one node to another. Unlike the free-flight operation, AirMatrix-based operation is one type of trajectory-based operation in which AVs will follow a set of waypoints with certain movement freedoms. In the proposed AirMatrix ConOps, AVs have enabled 26 movement freedoms from one node to another as shown in Fig. 3.2(b).



(a) Airspace structure



(b) Flight rule of UAV in AirMatrix airspace

Fig. 3.2: AirMatrix airspace structure.

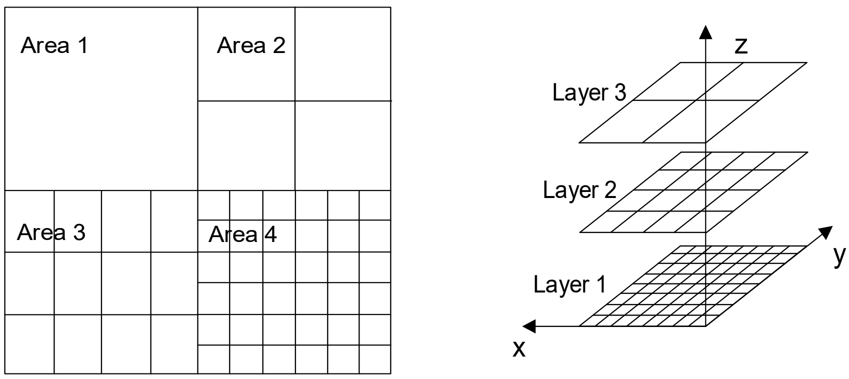
3.1.2 Adaptive Resolutions of AirMatrix

As aforementioned, the AirMatrix provides discrete and standard units of airspace, which is useful for uniformized airspace management. However, in some situations, uniformized air blocks will not be able to keep the operations safe and effective. In this regard, the different resolution of AirMatrix is proposed to balance the airspace complexity and traffic density.

The size of the air block significantly influences the airspace complexity and flight flexibility. For instance, with the decrease in air block size (Fig. 3.3(a) from Area 1 to

Area 4), the flexibility of flight will increase. Because in trajectory-based operational environments, Area 4 provides more waypoints for AVs to operate in the same area compared with Area 1, and the traffic density would increase accordingly from Area 1 to Area 4 if each air block serves the same number of operations. As the traffic density and flight flexibility both increase with the reduction of air block size, the airspace complexity will increase. With this relationship among air block size, flight flexibility, traffic density, and airspace complexity, different airspace areas can be designed with different resolutions. For instance, in densely populated areas, to ensure the safety of operation, the AirMatrix can be designed with a big size of air block to reduce the flexibility and complexity of airspace. While in sparse areas with less population, more flexibility can be given to achieve better efficiency.

Regarding three-dimensional airspace network design, the relationship between air block size and flight flexibility is illustrated in Fig. 3.3(b). The air block size reduces from Layer 3 to Layer 2, and Layer 1. For high-altitude layers, operations are more for high-speed, long-distance air taxi or air cargo travel, since high-altitude layers have fewer obstacles (e.g., buildings) to avoid. To this end, the main trunk route should be allocated for high-altitude layers to enable such kind of operations. While for very low altitude operations in urban environments (Layer 1), AVs need to avoid obstacles, taking off and landing frequently. In that case, more flight flexibility should be given, and the air block size should be small.



(a) In different flight areas (b) At various flight layers

Fig. 3.3: Flexible resolutions of AirMatrix Network.

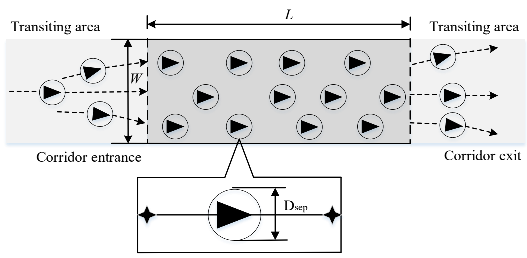
As depicted in Fig. 3.3(a), AirMatrix is formed by four different resolutions according to the respective population densities. For example, the parking area is less populated, while the industry zone has a medium population density. The residential area has a high population density and stations (MRT, shopping mall, hospital) have very high density. If the uniform resolution is applied in different population density areas, there will be unutilized nodes and links for parking and industry areas, wasting airspace resources. However, there would be insufficient flexibilities (nodes and links in the network) for high-demand areas such as a downtown area with dense high-rise buildings that need to be avoided.

3.1.3 AirMatrix Corridor

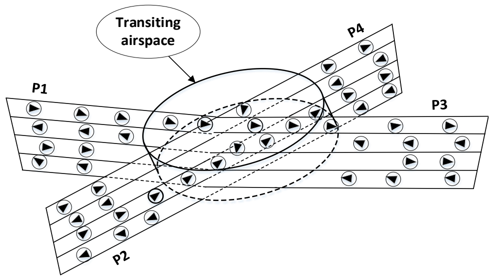
AirMatirx corridor serves as a bridge to connect separated areas with different resolutions and to provide high-speed traffic flow services, acting as a highway. The corridor plays two roles in contributing to safe and efficient operations. One is to provide an air tunnel where high-speed traffic flows are supported. In a completely covered large area, high-speed operations between separated areas are highly required. For instance, in the Singapore context, assuming AirMatrix can cover the whole country, operations between CBD, airport, and residential areas can be intense. And these operations are always time sensitive. Enabling high-speed traffic will be beneficial for these operations. Another role the corridor plays is to connect separated and reserved areas since not all areas are eligible to be covered by AirMatrix due to insufficient capacities in communication, navigation, and surveillance (CNS) or constraints (e.g., prohibited, restricted, and dangerous areas). Thus, the corridor can connect these separated areas and enable a smooth flow of AV traffic.

Each AirMatrix corridor has at least two layers. The traffic flows in these layers are separated laterally or vertically and it is one-way traffic (e.g., the traffic is only allowed for left-to-right operations in the same layer, see Fig. 3.4(a)). In the corridor, AVs will practice safety separation considering CNS performance. There are also transiting areas for AVs to change flight altitude and enter or leave the current corridor to smoothly join the next part. By doing so, the structure can mitigate risks from head-on collisions and will also improve efficiency by separating the traffic.

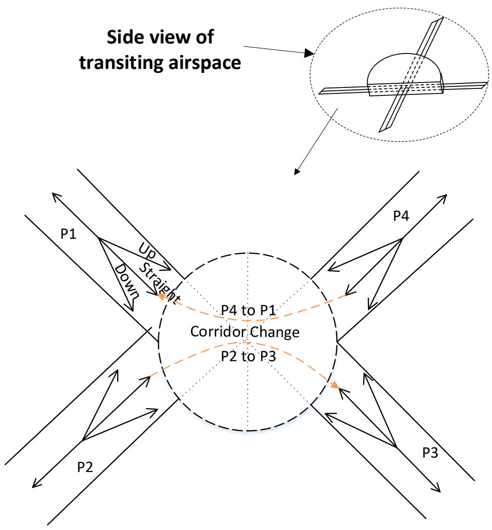
Fig. 3.4(b) shows the complex airspace with multiple crossing corridors and traffic flows. In each corridor, traffic is regulated in the corridor and cannot change the flight altitude. In transiting airspace, all traffic is allowed to change their flight altitudes. In Fig. 3.4(c), the side view of the complex airspace and operational rules is presented. P1 to P4 present the four corridors. Taking corridor P1 as an example, the climbing, descending, and straight flight in transiting areas are represented as left-turn, right-turn, and straight-flight in the Cartesian coordinate system, which helps for the kinematic analysis of complex traffic flows in corridor airspace.



(a) Traffic flow in AirMatrix corridor



(b) Corridor crossing



(c) Illustration of flight rules in the transiting airspace

Fig. 3.4: Concept of AirMatirx corridor and traffic flows. Note that in (c), the flight level change is illustrated up, straight, and down as shown in corridor P1. While the change among different corridors is illustrated with yellow dash lines from P4 to P1, and from P2 to P3.

3.1.4 Risk-Based Airspace Concept

Risk assessment and modelling is one of the key players in the role of aviation safety. Strategic risk assessment and risk-based route planning can significantly prevent safety issues from happening. In this section, the airspace risk assessment will be discussed under the framework of the AirMatrix.

This section will describe the risk cost map. Ground risks (UAVs hit pedestrians and vehicles on the ground) and air risks (UAVs hit manned aircraft) will be discussed. The modelling for each of these risks will be analyzed and incorporated into the risk cost map.

The airspace risk cost map is illustrated in Fig. 3.5. The map is formed in AirMatrix. Each link has a risk cost value computed according to the risk severity within the specified area. The color of the air block represents the risk cost value, in which different colors stand for different levels of risk severity. The route shown in Fig. 3.5 is the one avoiding the high-risk cost area (red color). It is obvious to see that the route is not the shortest-distance one, but its risk cost is much lower than the one entered the high-risk area.

The key point of this part is to illustrate the risk cost map and to propose risk assessment views. Ground risk and air risk are considered in the risk assessment process.

1) Ground people risk. As UAVs may lose control or power when flying and falling UAVs may hurt ground pedestrians. The risk cost of UAV hitting pedestrians is modelled as follows: There are three steps of a crash incident that will cause damage to pedestrians: (a) malfunction of UAV; (b) the falling UAV impacting pedestrians on the ground; and (c) the fatality damage caused to the pedestrian.

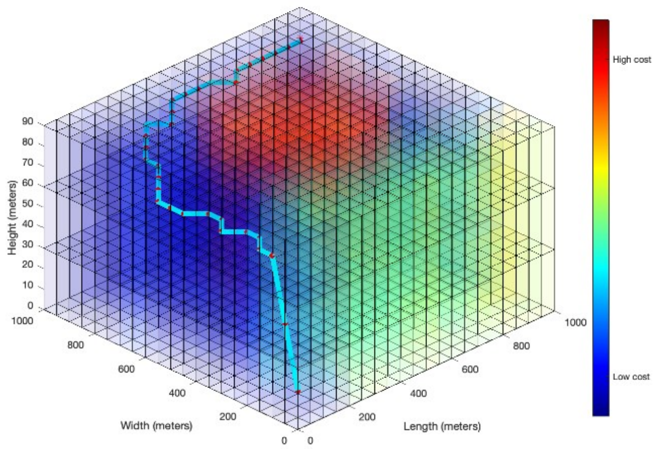


Fig. 3.5: Illustration of risk cost map and risk-based route planning.

2) Ground vehicle risk. There are also three factors when UAV causes an incident in road traffic: (a) UAV crashes above the road traffic; (b) UAV hits the vehicle on the ground; (c) crashed UAV causes a traffic accident and subsequent injuries to people.

Such as the ground pedestrian model, the expected fatality of a UAV impacting a ground vehicle can be defined as the number of vehicle accidents per hour caused by falling UAVs.

3) Air risk presents the collision risk between UAVs and manned aircraft. The safety issues of collision between manned aircraft and UAVs are crucial to the UTM-ATM integration. To this end, the present work also considers airports as risk areas where the risk should be accounted for. The incidence of collision between manned aircraft and UAV can be defined as the number of collisions per flight hour of UAV.

To safely and efficiently utilize urban airspace, several concepts of operations (ConOps) about urban airspace utilization are proposed and demonstrated in this work. The proposed ConOps concentrates on airspace configuration and operational rules from three aspects. Firstly, AirMatrix airspace configuration is introduced and expanded with operational rules in the network. To balance flight flexibility and airspace complexity, a different resolution AirMatrix is introduced. Moreover, the AirMatrix corridor is proposed to connect reserved areas with safe and efficient traffic flows considering Communication, Navigation, and Surveillance (CNS) performance. Secondly, Free-Flight Operation (FFO) and Trajectory-Based Operation (TBO) are illustrated and compared in terms of operational efficiency and airspace capacity based on simulation studies. Thirdly, under the AirMatrix framework, airspace risk assessment and contingency management are investigated to provide suggestions for urban airspace safety management and fail-safe system design.

3.2 Separation Requirement of AirMatrix under Trajectory Deviations

With the fast development of unmanned aircraft vehicle (UAV) operations in urban environments, low-altitude air traffic is projected to be crowded. That would arouse the problem of how to determine the safe separation between two UAV operations in structured airspace, as shown in Fig. 3.6. Similar to road traffic, how to determine the separation between each traffic lane for air traffic is also critical. It turns out to be a trade-off between safety and efficiency. Too wide a lane is a waste of land or airspace resources, while too narrow will have safety risk issues, as drones in the adjacent lane may collide with each other. Separation requirement between UAV operations becomes an essential challenge needing to be solved to ensure safety and improve the efficiency of urban low-altitude air traffic.

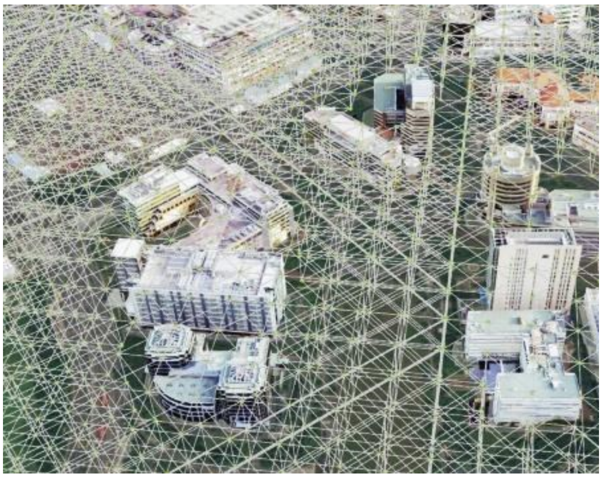
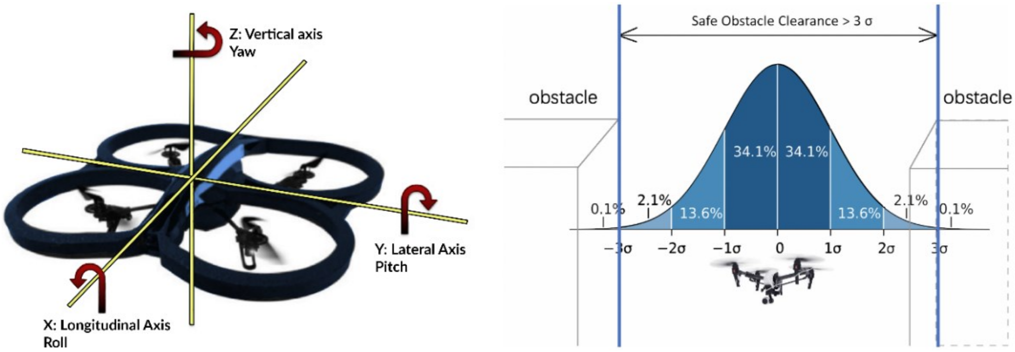


Fig. 3.6: AirMatrix airspace concept.

As aforementioned, the size of the standardized airspace cube can significantly influence the safety and efficiency of UAV operations. This work therefore will investigate the size of the airspace cube using a statistical method based on historical flight trajectory data. Specifically, this work will investigate two main problems: (i) Are the means and variances of UAV trajectory deviation in different flight timings statistically the same at a certain significance level. (ii) with the increase in flight height, is there any trend of the mean trajectory deviation.

Here, we define UAV trajectory deviation as the spatial deviation between the actual trajectory and the pre-planned trajectory. And we will investigate three axes of UAV operation, namely lateral, longitudinal, and vertical axis (Fig. 3.7(a)). That is corresponding to the size of the length (l), width (w), and height (h) of the airspace cube. To achieve this, one way is to obtain the standard deviation σ of specific drone operations, which can be obtained by analyzing historical flight data. For instance, we would like to achieve a drone flying within the airspace cube 99% of the time. The minimum obstacle clearance can be therefore obtained as $2*3\sigma=6\sigma$ (Fig. 3.7(b)) [46].

To get the trusted standard deviation σ, we need to study if the means and variances of trajectory deviation are the same for various operations in different flight times. We will conduct multiple flight experiments and collect flight data for analysis.



(a) Three axes of drone[1] (b) Correlation between cube size and deviation

Fig. 3.7: Illustration of how to determine the separation in three axes [46].

3.2.1 Separation Requirement Concerning Flight Technical Error (FTE)

As shown in Fig. 3.8, the workflow of this work starts with data input. There are five days of data are input and each dataset includes UAV trajectory data with more than 50 data points. Each data point is taken as the average trajectory deviation in three seconds. We conduct the normality test for obtained trajectory deviation data before proceeding to hypothesis testing and regression analysis.

The normality test data will be used in two branches. First, we use the data to conduct hypothesis testing with null hypothesis H0 and alternative hypothesis H1. The null hypothesis is presented as the population means from five testing days' data are statistically the same. While the alternative hypothesis is not all the population means are the same. If H0 is accepted, meaning that there is statistically significant evidence that all flight experiments have the same standard deviation. If reject H0, we perform a post hoc test to see which group(s) has a different σ causing the rejection. After identifying the abnormal observation(s), we can remove these outliers and continue to test the remaining samples. After iteration, we finally get the trusted standard deviation σ for cube size computation.

Another branch is regression analysis. We first compute the predicated parameters for both linear trends and quadratic trends, and then we conduct the test to see if the two

[1] https://www.oreilly.com/library/view/building-smart-drones/9781788477512/01335d5a-7ab3-4c8e-b1e8-9604821f5071.xhtml

trends are significant. Next, we will compute the measure of association strength to see which trend is more dominant. With all that done, we can conclude the two questions we investigate in this work.

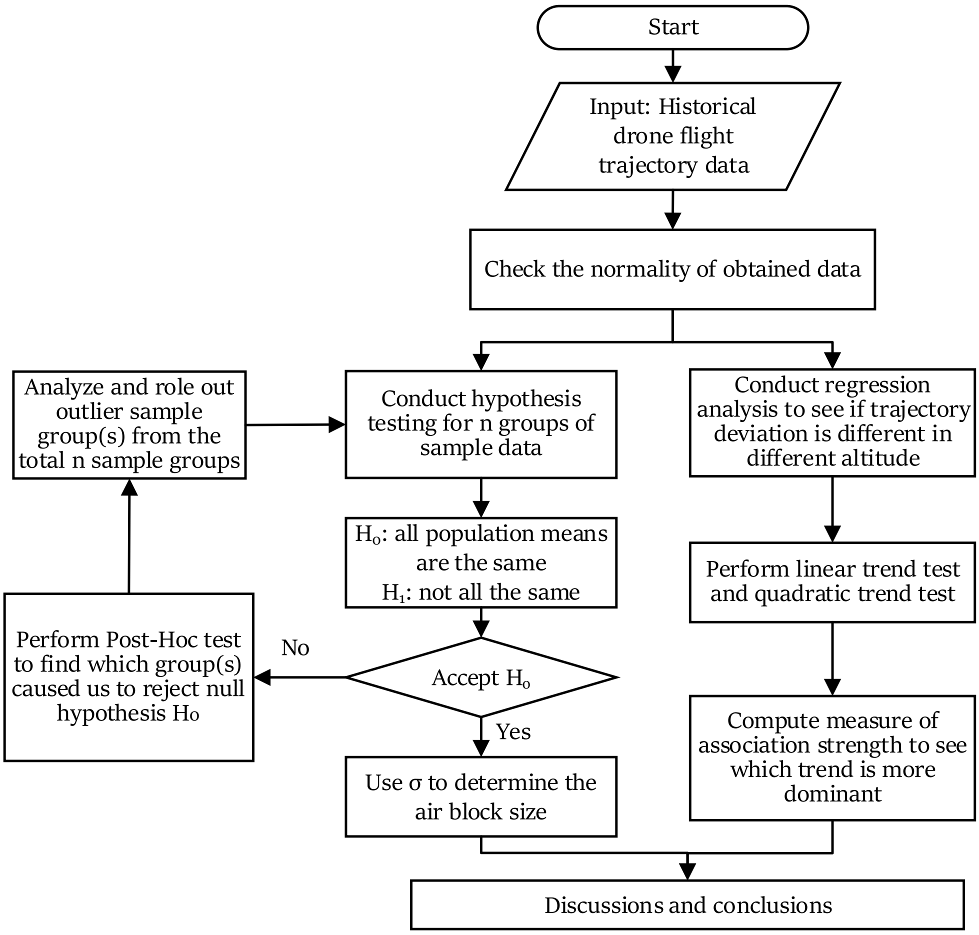


Fig. 3.8: Framework for determination of UAV separation requirements.

3.2.2 Hypothesis Testing for Trajectory Deviations Due to FTE

For the hypothesis testing part, there are two independent variables: flight testing time and flight height. The dependent variable is the UAV flight trajectory deviation. They are presented as follows.

Independent variables:

(1) Testing time: Day 1, Day 2, Day 3, Day 4, and Day 5.

(2) Flight height: 10 meters, 15 meters, and 20 meters.

Dependent variable:

(1) UAV flight trajectory deviation from reference path.

We define μ_{i_j} as the sample mean for ith day of j flight height. For instance, μ_{1_10} means the trajectory deviation mean of Day 1 at a flight height of 10 meters. Similarly, σ_{i_j} is the variance of trajectory deviation for ith day of j flight height. So, the hypothesis testing for the effect of different data of flying on trajectory deviation can be presented as:

Hypothesis (1): for population means

H_0: $\mu_{1\_10} = \mu_{2\_10} = \mu_{3\_10} = \mu_{4\_10} = \mu_{5\_10}$

H_1: Not all the means are the same

Hypothesis (2): for population variances

H_0: $\sigma^2_{1\_10} = \sigma^2_{2\_10} = \sigma^2_{3\_10} = \sigma^2_{4\_10} = \sigma^2_{5\_10}$

H_1: Not all the variances are the same

We will conduct the model comparison analysis to test the proposed hypotheses.

3.2.3 Regression Analysis on The Trend of Trajectory Deviations

The second part of this study is to conduct regression analysis to see if there is a significant trend of trajectory deviation with the increase in flight height. The independent variable is flight height, which has three choices of 10 meters, 15 meters, and 20 meters. The dependent variable is the flight trajectory deviation. They are presented as follows.

Independent variables:

(1) Flight height (X): 10 meters, 15 meters, 20 meters.

Dependent variable:

(2) Trajectory deviation (Y) from reference path.

We conduct linear trend analysis and quadratic trend analysis. And the hypothesis test will also be performed to see if the trends are significant.

(1) The linear trend model and hypothesis are given as

$Y = \beta_0 + \beta_1 x$

H_0: $\beta_1 = 0$

$H_1: \beta_1 \neq 0$

(2) The quadratic trend model is presented as

$$Y = \beta_0 + \beta_1 x + \beta_2 x^2$$

After the computation of predicted parameters and hypothesis testing, the measure of association strength for each trend will also be computed to see which trend is more prominent.

3.2.4 Flight Experiment Design and Numerical Analysis

A. Flight experiments and data collection

Five groups of sample data are obtained by drone experiment flying at different timings on the NTU campus. The experiment is performed with three flight heights (10m, 15m, and 20m), and the same experiment is conducted on five different days. The obtained actual flight trajectory will be processed based on the reference trajectory, which is a pre-planned path that a UAV is supposed to follow. After processing, we obtain the UAV flight trajectory deviation data for each testing day and at different flight heights.

B. Normality test for trajectory deviations

Deviations in the lateral axis

The raw data is obtained after processing, and they are provided in Appendix A. There are five days of testing data, and each day has three flight heights. We compute the mean and variance of these data and obtained results are given in Table 3.1.

With obtained data, we conduct the normality test using the *Lillietest function*. MATLAB software is used in this analysis study to perform tests. We call this function and load our data. The obtained histogram and Quantile-Quantile plot for normal distribution is given as Fig. 3.9. We also compute the statistical values for this example data, and obtained p_value is 0.4737, which is significantly greater than 0.05. So, we can conclude that the selected example data is normally distributed at a 0.05 significance level.

We conducted the same analysis for all five days' trajectory deviation data in different flight heights and different three axes. The trajectory deviations in different axes are significantly different due to the flight dynamics (see Fig. 3.7(a)). So, we perform the normality test for obtained data with three groups: lateral axis, longitudinal axis, and vertical axis group. The obtained results are shown in Table 3.2.

Table 3.1: Means and deviations of obtained sample data.

Test day	Flight height	**Lateral axis**		**Longitudinal axis**		**Vertical axis**	
		Mean (m)	Var. (m)	Mean (m)	Var. (m)	Mean (m)	Var. (m)
Day 1	10 m	5.0306	0.0379	3.6234	0.0647	0.1248	0.0058
	15 m	5.1346	0.0575	3.6554	0.0736	0.1623	0.0139
	20 m	5.4006	0.1089	3.7314	0.0804	0.1892	0.0151
Day 2	10 m	5.0386	0.0408	3.6260	0.0638	0.2015	0.0076
	15 m	5.1526	0.0569	3.6720	0.0699	0.1663	0.0065
	20 m	5.4086	0.0744	3.7280	0.0779	0.2193	0.0082
Day 3	10 m	5.0406	0.0439	3.6060	0.0794	0.0959	0.0040
	15 m	5.1306	0.0562	3.6580	0.0890	0.0988	0.0028
	20 m	5.4186	0.0944	3.7220	0.0830	0.1104	0.0020
Day 4	10 m	5.0546	0.0555	3.6434	0.0787	0.1698	0.0059
	15 m	5.1366	0.0689	3.6914	0.0951	0.2254	0.0048
	20 m	5.3666	0.1193	3.7514	0.1379	0.4228	0.0357
Day 5	10 m	5.0366	0.0695	3.6060	0.0696	0.1974	0.0166
	15 m	5.1786	0.0715	3.6040	0.0877	0.0734	0.0014
	20 m	5.3866	0.1586	3.6920	0.0896	0.4267	0.1136

As we can see from the results for the lateral axis, all data has p_value>0.05, except for the trajectory deviation data on Day 4 at 15 meters of flight height in the lateral axis. The p_value for it is 0.0237<0.05. This part of the data does not follow normal distribution at a 0.05 significance level. We remove this part of the data for the following analysis.

Deviations in the longitudinal axis

In this part, we test the normality for longitudinal group data. We employ the MATLAB *Lillietest function,* and the obtained results are given in Table 3.2. As we can see, the only dataset which has no significant normality is on Day 1 at 20 meters. The p_value for that dataset is 0.0226<0.05. So, we remove that dataset for the following analysis.

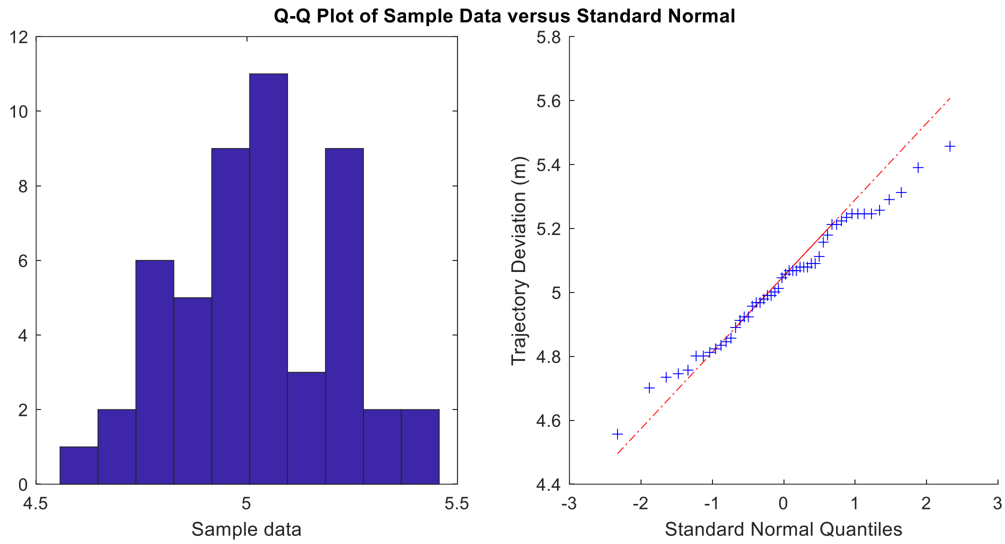


Fig. 3.9: One example of the histogram and Quantile-Quantile plot for normal distribution.

Table 3.2: P-values for three groups of experiment data (p>0.05~normal).

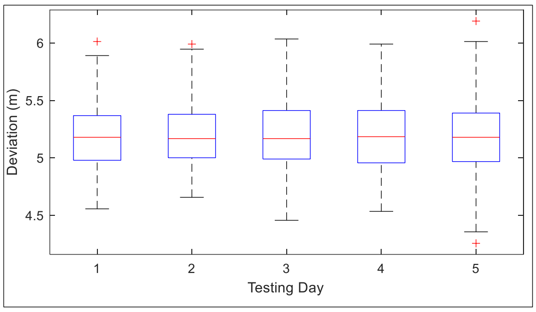
Axis	Lateral			Longitudinal			Vertical		
Timing and height (m)	10	15	20	10	15	20	10	15	20
Day 1	0.4737	0.5000	0.5000	0.3470	0.5000	0.0226	0.0448	0.0023	0.0010
Day 2	0.3936	0.5000	0.5000	0.5000	0.5000	0.1262	0.0323	0.2693	0.1317
Day 3	0.3807	0.2908	0.1641	0.1298	0.2987	0.3016	0.0149	0.1391	0.1216
Day 4	0.4767	0.0237	0.1611	0.5000	0.1246	0.5000	0.2907	0.0010	0.1938
Day 5	0.3246	0.5000	0.2977	0.5000	0.5000	0.3988	0.0082	0.0126	0.0038

Deviations in the vertical axis

We also test the normality for vertical group data. Quite differently, the vertical group data shows significantly less normality. This is because UAV flight dynamics, and in the vertical axis its performance is more stable and reliable, thus has very few deviations in the vertical axis. The mean vertical deviation is around 0.2 meters, which is significantly smaller than the deviations in the other two axes. So, in the following hypothesis testing and regression analysis, we will not involve the vertical group data.

C. Homogeneity test for the variance of trajectory deviation

Above we have tested the normality of all three groups' data. In this part, we check if all obtained trajectory deviation data follow the normal distribution with the same population variance. To do that, the *Vartestn function* is used, and the obtained test results are given in Fig. 3.10.



Group	Count	Mean	Std Dev
1	150	5.1886	0.30272
2	150	5.19993	0.28403
3	150	5.1966	0.30017
4	150	5.18593	0.31258
5	150	5.2006	0.34541
Pooled	750	5.19433	0.30965

Bartlett's statistic	6.3741
Degrees of freedom	4
p-value	0.1729

Fig. 3.10: Results of homogeneity test for the variance of trajectory deviation.

As P_value=0.1729>0.05, we can conclude that the variance among all obtained data has no significant difference at a 5% significance level.

D. Hypothesis testing

In this part, we conduct one-way ANOVA to test if the trajectory deviations have a significant difference between five different testing days. We use MATLAB ANOVA tool to conduct the analysis.

(1) Hypothesis (1): for population means at a flight height of 10 meters.

H0: $\mu_{1\_10} = \mu_{2\_10} = \mu_{3\_10} = \mu_{4\_10} = \mu_{5\_10}$

H1: Not all sample means are the same

(2) Hypothesis (2): for population means at a flight height of 15 meters.

H0: $\mu_{1\_15} = \mu_{2\_15} = \mu_{3\_15} = \mu_{4\_15} = \mu_{5\_15}$

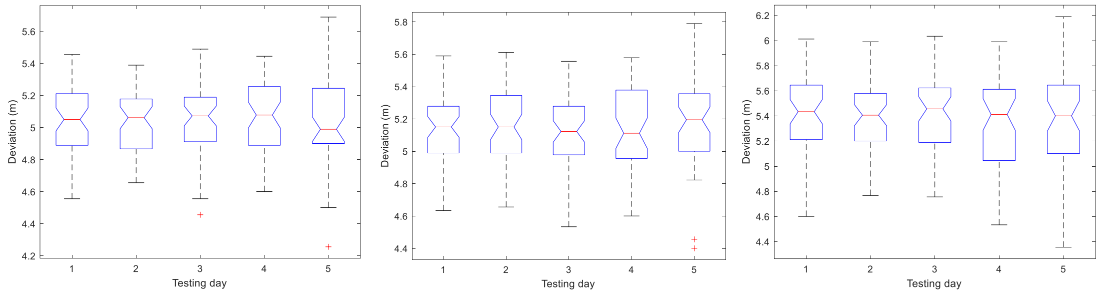
H1: Not all sample means are the same

(3) Hypothesis (3): for population means at a flight height of 20 meters.

H0: $\mu_{1\_20} = \mu_{2\_20} = \mu_{3\_20} = \mu_{4\_20} = \mu_{5\_20}$

H1: Not all sample means are the same

After computation, the p-values at different flight heights are p_10=0.9885, p_15=0.8688, and p_20=0.9461. As all p_values are greater than 0.05, we accept the null hypotheses that the population means in different testing days has no significant difference at a 0.05 significance level. The homogeneity test is also performed for the three groups of data, and there is no significant difference in variances between different samples. The results are presented in Fig. 3.11.



(a) 10 meters (b) 15 meters (c) 20 meters

Fig. 3.11: Results of homogeneity test for sample variance.

We also conducted the power of the test for these three hypotheses, and the observed power for lateral deviations is 0.66, while for longitudinal and vertical deviations are 0.53 and 0.58. As we can see the powers are relatively not satisfactory, and we can increase the power by adding more sample points.

E. Regression analysis for different flying height

In this part, we conduct the regression analysis to see if the trajectory deviation means have a trend with the increase in flight height. As discussed above, we only study the lateral group and longitudinal group data. As vertical deviations are small enough to be negligible. In regression analysis, the independent variable is flight height, which are 10, 15, and 20 meters. While the dependent variable is the mean of flight trajectory deviation. The independent and observed dependent variables used in the regression analysis are given in Table 3.3.

(1) Independent variable (x)= flight height (m)

(2) Dependent variable (y)= observed trajectory deviation

First, we conduct linear trend analysis and hypothesis test, and the model is presented as

Linear trend analysis:

$Y = \beta_0 + \beta_1 x$; H_0: $\beta_1 = 0$, H_1: $\beta_1 \neq 0$

Regression analysis for lateral trajectory deviation

Based on the regression model, we use the MATLAB regression function to compute the lateral trajectory deviation trend parameters. We have $\beta_0 = 4.6603$, $\beta_1 = 0.0356$. Thus, $y = 4.6603 + 0.0356x$.

Table 3.3: Independent and observed dependent values for regression analysis.

Axis	Lateral			Longitudinal		
x (m)	10	15	20	10	15	20
y (m)	5.0402	5.1466	5.3962	3.621	3.65615	3.725

To test if the linear trend is significant, we perform the hypothesis test with the null hypothesis H0: β_1=0, while the alternative hypothesis is H1: β_1≠0. The obtained results are:$R^2 = 0.9488$, and $F_{obs} = 18.5411 > F_{(0.05,1,150-3)} = 3.84$.

Based on the results, we can conclude that the linear trend is significant at a 0.05 significance level. We further conduct the quadratic analysis and obtained the predicted parameters as $y = 5.2570 - 0.0503x + 0.0029x^2$

Finally, we compute the measure of association strength for both linear and quadratic trends, and the results are given below. Based on the computed measure of association strength, we can see that the quadratic trend is more dominant, though not strong, for the sample data with $R^2_{alerting}$ is 55.69%. This is also more evident than the linear trend for these data with 44.31% of $R^2_{alerting}$.

	SS	R^2
Linear	101.881	0.4431
Quadratic	128.023	0.5569

We plot the predicted means based on a model that includes both linear and quadratic trend components, presented in Fig. 3.12. In general, the estimated means have a significant linear trend as we can see from the Figure below. The blue line is the estimated linear trend. The predicted means corresponding to 1st and 3rd heights are significantly close to the actual sample means. While for the 2nd height, the predicted mean is greater than the actual sample mean. For the quadratic trend, we can see that the means are significantly close to the actual means for all three points. The deviation between quadratic means and actual sample means is not significant.

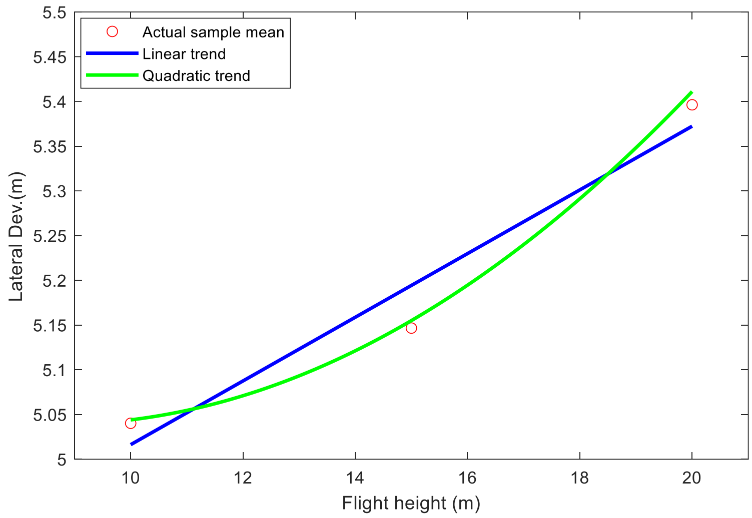


Fig. 3.12: Predicted means for both linear and quadratic trends in the lateral axis.

Regression analysis for Longitudinal trajectory deviation

We conduct the same regression analysis for longitudinal trajectory deviation. The procedures are similar to the abovementioned lateral analysis. We obtained the predicted parameters as $\beta_0 = 3.5114,\ \beta_1 = 0.0104$. Thus, we have the predicted linear trend model $y = 3.5114 + 0.0104x$. The other obtained results are: $R^2 = 0.9662$ and $F_{obs} = 28.5712 > F_{(0.05,1,150-3)} = 3.84$.

Based on the results, we can conclude that the linear trend is significant at a 0.05 significance level. We further conduct the quadratic analysis and obtained the predicted parameters as $y = 3.6518 - 0.0098x + 0.0007x^2$.

Finally, we compute the measure of association strength for both linear and quadratic trends, and the results are given below. Based on the computed measure of association strength, we can see that the linear trend is more dominant, though not strong, for the sample data with $R^2_{alerting}$ is 53.03%. This is also more evident than the linear trend for these data with 46.97% of $R^2_{alerting}$.

	SS	R^2
Linear	6.511	0.5303
Quadratic	5.768	0.4697

We plot the predicted means based on a model that includes both linear and quadratic trend components, presented in Fig. 3.13. In general, the estimated means have a significant linear trend as we can see from the figure below. The blue line is the estimated linear trend. The predicted means corresponding to 1st and 3rd heights are significantly close to the actual sample means. While for the 2nd height, the predicted mean is greater than the actual sample mean. For the quadratic trend, we can see that the first mean is significantly close to the actual mean, while the other two predicted means are not well close to the actual means.

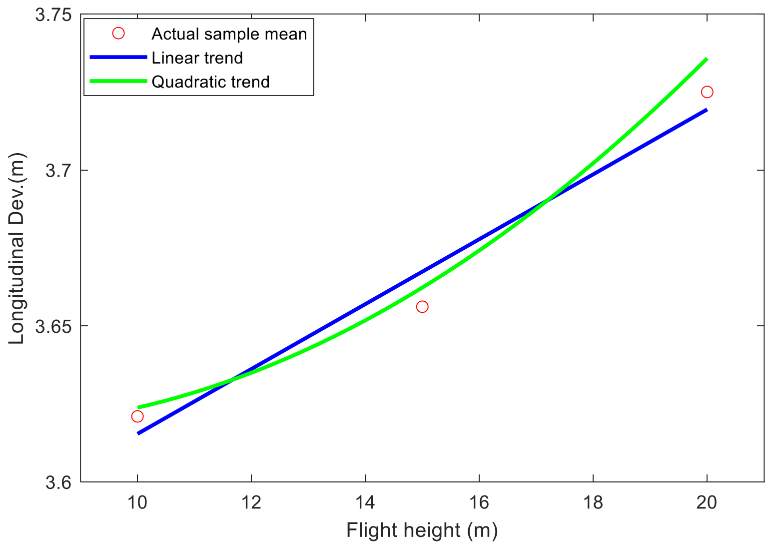


Fig. 3.13: Predicted means for both linear and quadratic trends in the longitudinal axis.

F. Application of tested results in air block size determination

After the above tests and analysis, we know that the obtained trajectory data follow the normal distribution at a 0.05 significance level, while the obtained data on different testing days and different flight heights have the same means and variances at a 0.05 significance level. So, we can say that the UAV flight trajectory deviation is consistent in different flight times and flight heights. We can use the obtained standard deviation to determine the size of the air block. Based on the analysis in Fig. 3.7(b), we know the size of the air block in one particular axis is six times its standard deviation (6σ) in that axis at a 0.01 significance level. Therefore, we can have the following results shown in Table 3.4. The safe obstacle clearance means that UAVs operating in such size as an air block will 99% of the time does not fly out of the air block. This size also does not consider any safety buffer and protection zone for the air traffic corridor.

Table 3.4: Estimated safe obstacle clearance at 0.01 significance level.

Parameters	Lateral	Longitudinal
Variance (m)	5.19	3.67
Standard trajectory deviation (m)	2.28	1.92
Safe obstacle clearance (m)	14	11

This work studies a statistical method for the determination of air block size to make a better trade-off between the safety and efficiency of drone operations. Hypothesis testing is conducted to test the means and variances of sample data. The regression analysis is also performed to investigate the trend of trajectory deviation with the increase of the flight height. The main findings of this part can be concluded as follows.

(1) The UAV flight trajectory deviation at the same height follows normal distribution when the sample size is 50 at a 0.05 significance level. While the trajectory deviation in different testing data and different flight heights also follow the normal distribution, which has the same variance.
(2) The means of trajectory deviation in different testing days are statistically the same at a 0.05 significance level. This indicates that flight timing has no significant impact on the mean trajectory deviations.
(3) With the increase in flight height, the trajectory deviation of the drone will increase. The trajectory deviation will become significantly larger in both lateral and longitudinal axes when UAV flies higher. The linear increasing trend is more evident in lateral deviation while the quadratic trend dominates in the longitudinal deviation.

The research findings of this part rely on the obtained experimental data in a specific environment we selected. In future works, more experiments can be conducted with other types of UAV and flight environments to collect finer sample data used for UAV separation requirement analysis. Moreover, separation requirements for flights with payloads (e.g., parcel delivery cases) can also be investigated, as the payloads may serve as a damper to mitigate the deviation effect or act as a booster to amplify the deviation due to inertia effects. Note that different types of UAVs in different environmental conditions may yield different results. More flight experiments and testing are still required to understand different flight technical errors to support the investigation of UAV separation requirements. Wind information plays a significant role in trajectory deviation analysis [144], which should be considered and incorporated into trajectory deviation analysis in future studies. Besides, with more data becoming available for more types of UAVs, the methodology of trajectory deviation analysis can be extended to more AAM operations.

3.3 Research Strategies with AirMatrix Concept

This Chapter proposes concepts of operations (ConOps) for UAS operations in urban airspace. The proposed ConOps focuses on risk-based airspace configuration and rules.

Firstly, AirMatrix airspace configuration is introduced and expanded with operational rules in the network. To balance flight flexibility and airspace complexity, a different resolution AirMatrix is introduced. Moreover, the AirMatrix corridor is proposed to connect reserved areas with safe and efficient traffic flows considering Communication, Navigation, and Surveillance (CNS) performance.

Secondly, free-flight operation (FFO) and trajectory-based operation (TBO) are illustrated and compared in terms of operational efficiency and airspace capacity based on simulation studies. Thirdly, under the AirMatrix framework, airspace risk assessment and contingency management are investigated to provide suggestions for urban airspace safety management and fail-safe system design.

In the AirMatrix concept, one important question is to determine the size of the air block, which is to determine the safe separation between two UAV operations. This chapter proposes an evaluation framework for the determination of flight technical error (FTE) by investigating flight trajectory deviations, which is to support the establishment of separation requirements for safe UAV operations in low-altitude urban airspace under AirMatrix concepts.

The studies in this chapter prepare the fundamental concepts of operations and operational environments for third-party ground risk assessment and mitigation in Chapter 4 and Chapter 5, and air risk mitigation in Chapter 6 and Chapter 7.

Chapter 4

4 Ground Risk Modelling and Mitigation

In this Chapter, ground risk assessment and mitigation problems are solved by a novel unmanned aerial vehicle (UAV) flight path optimization method that considers an integrated cost assessment model. The assessment model incorporates fatality risk, property damage risk, and noise impact, which is an extension of current third-party risk indicators at modeling and assessment levels. To solve the proposed optimization problem, a hybrid estimation of distribution algorithm (EDA) and CostA* (named as EDA-CostA*) algorithm is proposed, which provides global and local heuristic information for path searching in cost-based environments. Simulation results demonstrate the effectiveness and reliability of the developed optimization model in reducing risk costs.

4.1 Risk Modelling and Assessment

In this section, the third-party risk models are proposed with the introduction of new risk sources of societal impacts on the public. An integrated risk assessment model is developed to capture risk factors and identify high-risk areas for risk-based route planning problems.

4.1.1 Third-Party Risks in Urban Environments

In metropolitan environments, there are dense populations, high-rise buildings, critical infrastructure, etc. UAVs operating in such low-altitude airspace will encounter various risk issues [26]. In this Chapter, the scope of operation altitude is below 400 feet above ground level [145]. Recent studies investigated various risks in urban environments, and most of their attention is on the risks of the impact on ground people and vehicles [10], [146], midair collision with small UAVs and manned aircraft [46], [147], [148], noise impact [12], as well as UAV operational cost and efficiency [149]. We conclude the primary risk and impact factors, as illustrated in Fig. 4.1.

(1) Fatality risk. Crashed UAV impacts pedestrians and vehicles on the ground, causing fatalities on people.
(2) Property damage risk. Crashed UAV hits critical infrastructures or collides with high-rise buildings, causing property loss.
(3) Noise impact. Noise impact on the public is a big concern for the acceptance of UAV operations in urban environments, which will be assessed and mitigated.

The risk factor of a midair collision between UAV and manned aircraft in urban airspace [48] is not considered in this work, as the aerodrome control zone is treated as a restricted area where the UAV is strictly not allowed to enter. The risks of UAVs intruding on military-related bases and facilities are also out of the scope of this study.

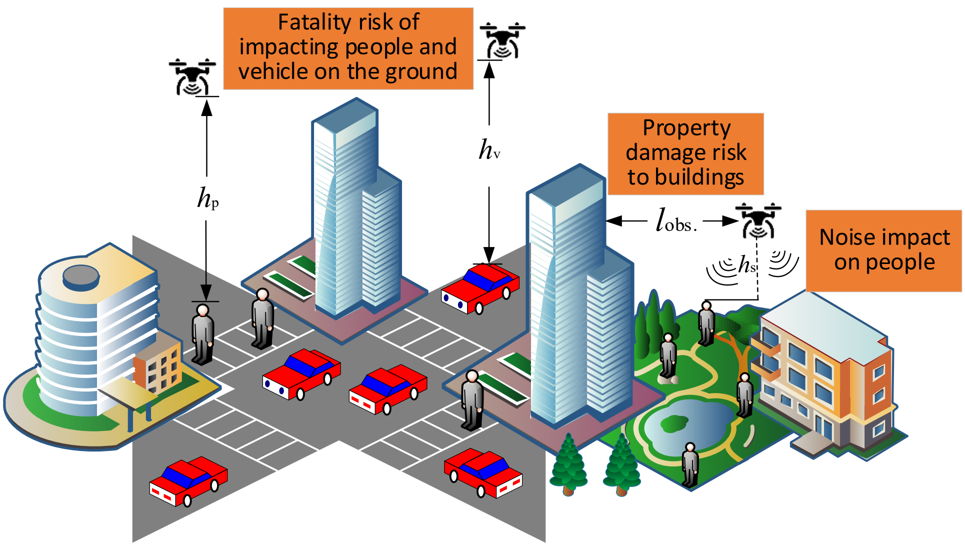
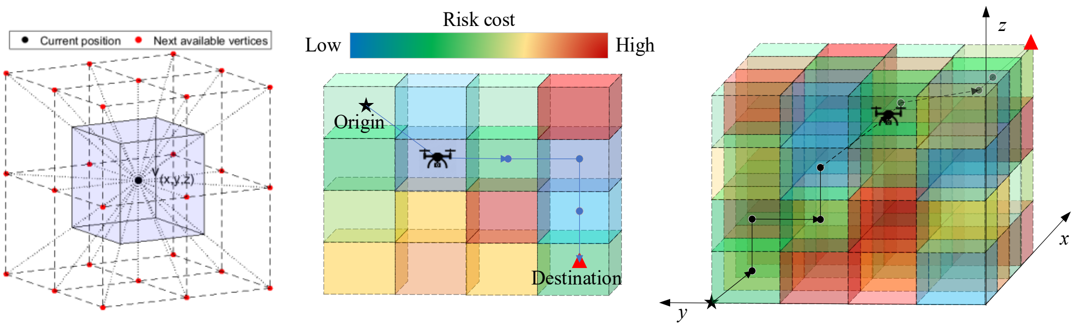


Fig. 4.1: Illustration of fatality risk, property damage risk, and noise impact of UAV operations in metropolitan environments.

The risk factors of population density, vehicle density, and buildings are discretely distributed in metropolitan environments. Subdivision of airspace into smaller units enables more flexible management [150]. To quantitatively assess these risks, urban low-altitude airspace is divided into discrete three-dimension air block units and the centroid of the unit is denoted as v_{xyz} (Fig. 4.2(a)). UAVs have 26 possible choices to move from one vertex to another, and it follows the centroid point to maintain safe separation from other UAVs in adjacent air blocks. Risk cost assessment of each airspace unit is conducted based on its pertaining environments such as population density and vehicle density underneath. The risk is represented as colored air blocks (Fig. 4.2(b)), and the 3D cost-based flight path is illustrated in Fig. 4.2(c). UAVs operate in complex 3D risk maps to avoid high-risk areas (presented in red color) and to minimize total operational risk cost.



(a) 26 options from current position to next position

(b) Risk-based airspace and flight path

(c) 3D view

Fig. 4.2: Illustrations of risk-based airspace and flight path planning.

The core of the risk-aware airspace model and route planning is the accurate and quantitative risk value of each airspace unit. To achieve this, an integrated risk assessment model is proposed to compute the risk value.

4.1.2 Risk Modelling and Assessment

The integrated cost assessment model includes three parts: fatality risk model, property damage risk cost model, and noise impact cost model. We develop the corresponding models for the cost assessment.

4.1.2.1 Fatality risk model

Fatality risk is caused due to UAV crashes to pedestrians and persons in ground vehicles, and the unit is the number of fatalities per flight hour. The fatality risk is denoted as $c_{\mathrm{r\_f}}$, and it consists of the risk to pedestrians $c_{\mathrm{r\_p}}$ and to the person in the vehicle $c_{\mathrm{r\_v}}$.

$$c_{\mathrm{r\_f}} = c_{\mathrm{r\_p}} + c_{\mathrm{r\_v}} \tag{4.1}$$

A. Fatality Risk Model: Risk of UAV impacts people on the ground

UAVs may lose control and crash to pedestrians on the ground (see Fig. 4.1). To evaluate the fatality risk to pedestrians, there are three processes [22], [151] need to be considered: (1) system failure of UAV; (2) UAV crashes to pedestrians; and (3) fatal damages caused to the pedestrians. That falling UAV hits people on the ground causing fatalities is a sequential action, which is associated with the three processes mentioned above. The risk cost of UAV crashes to pedestrians $C_{\mathrm{r\_p}}$ is defined as the number of fatalities per flight hour, denoted as

$$C_{\mathrm{r\_p}} = P_{\mathrm{crash}} N_{\mathrm{hit}}^{\mathrm{p}} R_f^{\mathrm{p}} \tag{4.2}$$

where $c_{\mathrm{r\_p}}$ is the risk value associated with the fatality of pedestrians, and P_{crash} is the probability of UAV system failure. Note that $N_{\mathrm{hit}}^{\mathrm{p}}$ is the number of pedestrians hit by crashed UAVs (correlated with population density), and R_f^{p} is the fatality probability that people are killed in UAV accidents.

The P_{crash} is primarily determined by the reliability of the UAV system itself, including hardware and software reliabilities. The R_f^{p} is correlated with the weight and falling height of the UAV. The uncertain variable in Eq. (4.2) is $N_{\mathrm{hit}}^{\mathrm{p}}$, which associates with the population density, defined as

$$N_{\mathrm{hit}}^{\mathrm{p}} = S_{\mathrm{\_hit}} \sigma_{\mathrm{p}} \tag{4.3}$$

where $S_{\mathrm{\_hit}}$ is the size of crash impact area, and σ_{p} is the population density in the administrative unit.

The fatality probability R_f^{p} is associated with two main factors: impact kinetic energy and sheltering effects. The kinetic energy E_{imp} of falling UAV primarily determines the severity of impact, while the sheltering coefficient S_c affects the degree of impact on the people and vehicles, as the buffering effects of buildings and trees will soften the. The sheltering coefficient S_c is introduced as the absolute real number $S_c = (0,1]$ [152], and the fatality probability is presented as

$$R_f^p = \frac{1}{1 + \sqrt{\frac{\alpha}{\beta}} \left(\frac{\beta}{E_{\mathrm{imp}}} \right)^{\frac{1}{4S_c}}} \tag{4.4}$$

where α is the impact energy that might cause 50% fatality with $S_c = 0.5$, while β is the impact energy threshold required to cause fatality as S_c approaching zero (see Fig. 2 in (Dalamagkidis et al 2008)). Based on that, we take α =106 J and β =100 J.

The impact kinetic energy E_{imp} of the falling UAV is known as

$$E_{\mathrm{imp}} = \frac{1}{2} m v^2 \tag{4.5}$$

where m (kg) is the mass of the falling UAV, and v is the velocity when the UAV hits the ground. The processes to compute v will be presented below.

The vertical drag force F_d of falling UAV is related to its size and materials, as well as the density of air, etc., denoted by [30]

$$F_d = \frac{1}{2} R_I S_{\_\mathrm{hit}} \rho_A v_{\mathrm{TAS}}^2 \tag{4.6}$$

where R_I is the drag coefficient that is related to the UAV type, and ρ_A is the density of air (1.225 kg/m^3 at sea level), and v_{TAS} is the true airspeed of the falling UAV.

The acceleration of UAV can be then presented as

$$a = \frac{F_g - F_d}{m} = g - \frac{R_I S_{\_\mathrm{hit}} \rho_A v_{\mathrm{TAS}}^2}{2m} \tag{4.7}$$

where F_g is the gravitational force, $F_g = mg$. (g=9.8m/s^2)

Thus, the velocity v of UAV hits the ground at moment t can be obtained as

$$v = \int_0^t \left(g - \frac{R_I S_{\_\mathrm{hit}} \rho_A v_{\mathrm{TAS}}^2}{2m} \right) dt = \sqrt{\frac{2mg}{R_I S_{\mathrm{hit}} \rho_A} \left(1 - e^{-\frac{h R_I S_{\mathrm{hit}} \rho_A}{m}} \right)} \tag{4.8}$$

where h is the start falling height of the UAV above the ground surface.

B. Fatality Risk Model: Fatality risk to persons in the vehicle

Fatality risk to persons in the vehicle is modelled and there are also three components of a crash UAV accident on vehicles [32]: (1) UAV system failure; (2) falling UAV hits a vehicle; (3) the crash accident causes a direct fatality to people or causes a traffic accident which subsequently kills persons in the vehicle.

Fatality risk that UAV crashes to persons in the vehicles is defined as the number of fatalities per flight hour, denoted as

$$C_{\text{r\_v}} = P_{\text{crash}} N_{\text{hit}}^{\text{v}} R_f^{\text{v}} \tag{4.9}$$

where $N_{\text{hit}}^{\text{v}}$ is the number of vehicles hit by the crashed UAV, and R_f^{v} is the average fatality probability that persons in vehicles are killed in accidents, which can be obtained by: (annual total fatalities caused by car accidents)/(annual total number of car accidents).

The average number of vehicles that might be hit by crashed UAV can be defined as the ratio of the total projected area of all vehicles and the total road area, denoted as

$$N_{\text{hit}}^{v} = S_{\text{\_hit}} \sigma_v \tag{4.10}$$

in which σ_{v} is the traffic density in the administrative unit u.

C. Estimation of population density and traffic density using gravity model

The population density and traffic density distributions in metropolitan environments are the essential variables which will directly influence the UAV operational risk costs as discussed in Eqs. (4.3) and (4.10). Based on the previous studies, these density distributions are strongly correlated with the consumption amenities [153], which attracts people and vehicle. To quantitively assess this correlation between consumption amenities and population density, the gravity model was used to calculate the population density [154]. Inspired by the gravity model [52] and population mapping method [155], we have the following formulas to compute the population density in urban environments.

The population density of a unit u is given as

$$\sigma_{\text{p}} = e^{(1-r^2)} \sigma_{\text{p.avg}} \tag{4.11}$$

where $\sigma_{\text{p.avg}}$ is the average population density in the given area. Note that r is the radius of the gravity influence area induced by the amenity, which is given as 1 km in this work. As shown in Fig. 4.3, the population density decreases in an inverted exponential pattern with the increase of the radius r. In the first 0.3 km of r, the population density index remains high, which captures the high population density distribution in the vicinity of amenities. While in the range of 0.3 km to 1.0 km, the index drops linearly, presenting an even decreasing trend of population density.

Similarly, the road traffic density distribution can be denoted as

$$\sigma_{\text{v}} = e^{(1-r^2)} \sigma_{\text{v.avg}} \tag{4.12}$$

where $\sigma_{\mathrm{v.avg}}$ is the average traffic density in the given area.

4.1.2.2 Property Damage Risk Cost Model

Dense high-rise building in urban environments is another challenge for UAV operations. Potential collisions with buildings pose property damage risks, and densely distributed high-rise buildings also limit the speed of traffic flow, resulting in the inefficiency of the UAS system [156]. The property damage risk cost model also integrates the operational efficiency cost, which is accounted for the planning and optimization of airspace and traffic flow.

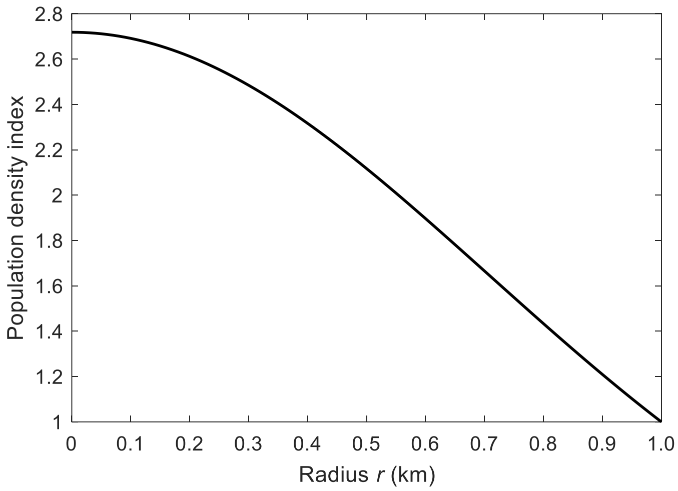


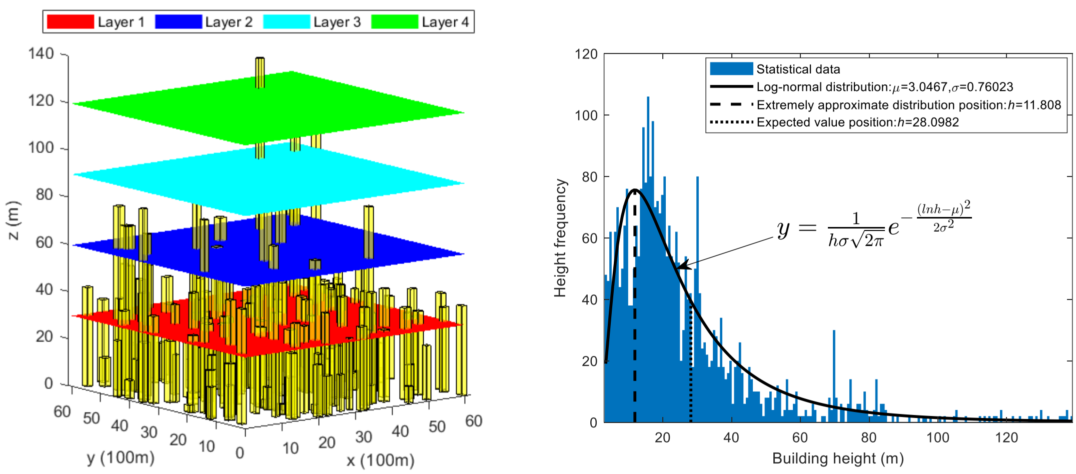
Fig. 4.3: Illustration of population density index changes with influence radius *r*.

The flight altitude is a primary variable of the property damage risk model. As Fig. 4.4(a) depicted, in the low-altitude layer (Layer 1, for instance), the density of the building is high. UAV operating in Layer-1-type airspace needs to frequently perform deconfliction to avoid obstacles, increasing risks and efficiency loss. In the high-altitude layer (i.e., Layer 4), in contrast, there are few buildings to affect UAV operations, so that operational safety and efficiency can be significantly improved. Here, we assume that the operation will be affected if the flight distance of a drone from buildings is smaller than the separation minima. The building density distribution at different heights plays a key role in property damage risk modelling, which is correlated with the collision probability of UAVs with buildings. The building height distribution is not fit with the standard normal distribution but the log-normal distribution [20], [21], height is a nonnegative value, and its distribution is not symmetrical (Fig. 4.4(b)). Based on these characteristics, the correlation between building height and property damage risk cost $c_{\mathrm{r\_p.d}}$ is established as

$$\psi(h;\mu,\sigma) = \frac{1}{h\sigma\sqrt{2\pi}} e^{-\frac{(lnh-\mu)^2}{2\sigma^2}} \tag{4.13}$$

$$c_{\mathrm{r\_p.d}} = \begin{cases} \psi(e^{\mu}), & 0 < h \le e^{\mu} \\ \psi(h), & h > e^{\mu} \end{cases} \tag{4.14}$$

where h is the building height, μ and σ are the mean and standard deviation of the logarithmic variable. Note that $c_{\mathrm{r\_p.d}}$ is the risk cost of property damage upon drone operation. For buildings with a height smaller than the threshold of e^{μ}, the risk cost equals the one which the height e^{μ} has Eq. (4.14) defines). This is because below that height ($h = e^{\mu}$), buildings are dense, and the risks are high. The biggest risk cost value is therefore taken as the one with height $h = e^{\mu}$. In this study, the building density and height data are obtained from a downtown area (size: 6km×6km) in Singapore. A total number of 3366 buildings are included in the selected area. By computing statistical features, the mean value for the log-normal distribution is obtained as μ =3.0467. While for buildings with a height greater than e^{μ}, the operational risk cost is computed as the log-normal distribution presents in Eq. (4.13). Meaning that with the increase in building height ($h > e^{\mu}$), the property damage risk cost decreases, as in higher layers there are fewer building obstacles to influence the safe operations of UAVs. Note that the property damage risk factor is to facilitate the determination of an optimal flight layer in particular areas.



(a) Illustration of building height distribution height

(b) Log-normal distribution of building

Fig. 4.4: Building obstacles impact on UAV operation in urban areas. In (a), the building height data is selected from a particular city area in Singapore, where 100 building clusters have been selected to illustrate the building height distribution. In (b), the statistical features of building height distribution are analyzed using log-normal distribution. Here, the height frequency presents the number of buildings at such height.

4.1.2.3 Noise impact cost model

Noise impact is another important societal issue and needs to be mitigated when UAVs operate in low-altitude urban environments [12]. That should be therefore considered as a cost when conducting UAV flight planning. The impact of noise on the public is in effect when UAVs operate close to people, especially at nighttime. While with the increase of flying altitude, the impact will decrease to the threshold which will not have effects for people on the ground. The correlation of noise impact cost and its flying height is illustrated in Fig. 4.5. As we can see from Fig. 4.5(a), the key factor of noise impact cost is UAV flying height, as the sound propagation is correlated with the height and distance of UAV away from the people.

An approximation of sound propagation is the spherical spreading, which is presented as

$$I(si) = \frac{1}{h^2 + d^2} \tag{4.15}$$

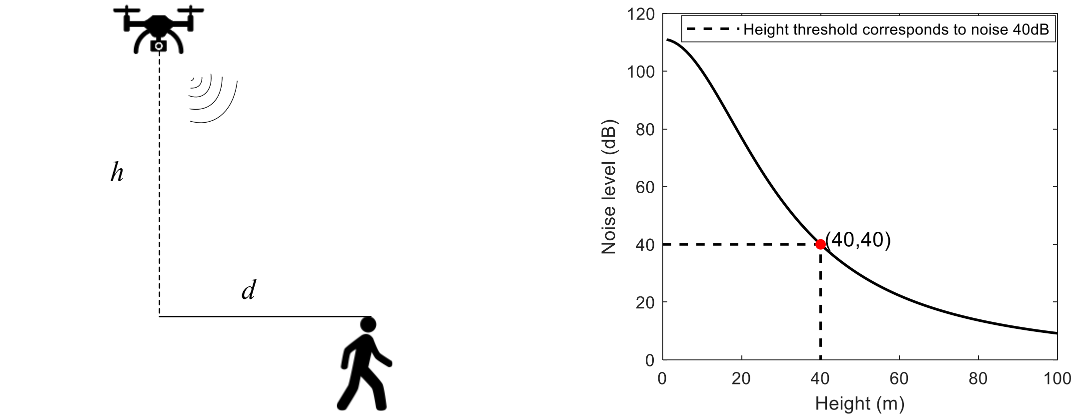
where $I(si)$ is the sound intensity at height h and distance d from the point directly under the UAV. Here d is taken as 30 feet [157]. The sound intensity is further converted as sound level,

$$L(sl) = \varpi \mathrm{L}_h I(si) \tag{4.16}$$

in which $L(sl)$ is the sound level (dB); ϖ is the conversion factor from sound intensity to sound level; L_h is the reference noise produced by a drone, taken as $\mathrm{L}_h = 55$dB [158].

$$c_{\text{noise}} = L(sl) = \varpi \mathrm{L}_h \frac{1}{h^2 + d^2} \tag{4.17}$$

where c_{noise} is the cost of noise impact upon drone operation in the given airspace unit. Noise impact will not be considered as a cost for UAV operation if flying height exceeds the threshold. Based on previous studies [59], [61], we take the height threshold as the one that corresponds to the noise level of 40 dB, illustrated in Fig. 4.5(b).



(a) Illustration of noise impact (b) Correlation of noise level and flying height

Fig. 4.5: UAV noise impact on people.

4.2 Risk-based UAV flight path optimization

In this Section, the cost-based path planning is modelled, and the solution algorithm is also developed. First, we propose an integrated cost model for the optimization of the UAV flight path considering fatality risk, property damage risk, and noise impact. Based on the integrated cost model, we formulate the risk cost-based path planning as a minimum cost flow problem in an undirected graph. To solve the formulated problem, we propose a hybrid algorithm that incorporates the estimation of distribution algorithm (EDA) and CostA* algorithm.

4.2.1 Integrated risk cost model

The total operational cost integrates fatality risk cost, property damage risk cost, and noise impact cost. As the weightage of these three types of cost might be different due to their significance or user preferences, the contribution of each type of cost will also be various. For instance, aviation regulators may take safety as the top priority, requiring a very low fatality risk cost of UAV operation. In this regard, the weight of fatality cost should be increased. By doing so, areas with dense populations and vehicles will be identified as high-risk areas by the proposed model, and the path planning will subsequently avoid these areas. To quantify the significance and preferences of UAS stakeholders on different risk types, the weight factor α_τ is introduced. As the three risk cost factors are exclusive and are normalized into the range of (0, 1], the integrated cost assessment model is developed as an additive linear model

$$c_v = \sum_{\tau=\{\text{r\_f},\text{r\_p.d},\text{noise}\}} \alpha_\tau \omega_\tau c_\tau \tag{4.18}$$

where c_v is the integrated cost of a vertex. Here α_τ is weight factor, ω_τ is a normalization factor and c_τ is the cost value. Note that $\tau = \{\text{r\_f}, \text{r\_p.d}, \text{noise}\}$ corresponds to fatality risk, property damage risk, and noise impact, respectively. Here $\{\alpha_\tau: \alpha_1, \alpha_2, \alpha_3\} \in [0,1]$, and $\alpha_1 + \alpha_2 + \alpha_3 = 1$. Note that $\{\omega_\tau: \omega_1 = \omega_{\text{r\_f}}, \omega_2 = \omega_{\text{r\_p.d}}, \omega_3 = \omega_{\text{noise}}\}$, and $\{c_\tau: c_1 = c_{\text{r\_f}}, c_2 = c_{\text{r\_p.d}}, c_3 = c_{\text{noise}}\}$.

The normalization factor ω_τ is used to eliminate the magnitude effect among different risk types [159]. The cost c_τ is divided by the maximum cost value c_{τ_max}. Here c_{τ_max} is defined as the maximum cost value among all vertices in a given environment. For example, the given environment has meshed into standard airspace units that have 100 vertices. The fatality risk is computed for all vertices and the maximum cost value is ten, for instance. Then, the fatality risk value of all 100 vertices will be divided by ten, and all the cost values will fall in the range of (0,1]. The correlation between the normalization factor and the maximum cost value is denoted as

$$\omega_\tau = \frac{1}{c_{\tau\_\text{max}}} \tag{4.19}$$

4.2.2 Problem formulation of cost-based path planning

To facilitate the cost assessment, airspace is divided into uniform three-dimensional units. These units consist of vertices and edges and can be represented as a graph. The cost-based path planning problem is a special case of the minimum cost flow problem, which is to find a path in the graph from origin to destination with minimum cost. We modelled the problem as follows:

Let $G = (V, E, C)$ be an undirected, connected, and weighted graph for which:

$V = \{v_1(x_1, y_1, z_1), v_2(x_2, y_2, z_2), \dots, v_n(x_n, y_n, z_n)\}, n \in \mathbb{N}$ presents the finite set of graph vertices of the divided airspace.

$E \subset V \times V$ denotes the finite set of edges connecting the vertices in the graph.

$C: E \to \mathbb{R}_0^+$ presents the weight function of the edge.

In this Chapter, the weight represents the cost of the edge. The cost of an edge $e_i(v_i, v_{i+1})$ can be denoted as c_{e_i}. The integrated cost is given to the vertex of the airspace unit centroid. To align with that, the cost of the edge c_{e_i} is represented by the cost of the vertex on the edge, denoted as c_{v_i}. The weight function of the edge is equivalent to the cost function of the vertex as shown in Eq. (4.18). Let s and t be distinct vertices of G. A path P from s to t in G is called the most cost-effective path if C_P is minimum among all paths from s to t in G.

The objective of this work is to optimize the integrated cost of the planned path, denoted as

$$\min: C_P = \sum_{v_i \in V_P} c_{v_i} \tag{4.20}$$

where C_P is the total cost of path P, and c_{v_i} is the cost of the air block that contains a vertex v_i. Note that V_P is the set of vertices on path P.

As Fig. 4.2 shows, 26 available vertices can be chosen as the next point to move. Specifically, six vertices are straightly connected with the current vertex, while 12 vertices are connected as planar diagonals and eight vertices are connected as cubical diagonals. To ensure the planned flight path is operationally valid in the sense that a complete sequence of waypoints is provided for autonomous navigation, the constraint of waypoint selection is presented as

$$\begin{cases} x_{i+1} = x_i + \delta_x^i \\ y_{i+1} = y_i + \delta_y^i \\ z_{i+1} = z_i + \delta_z^i \\ \delta_x^i, \delta_y^i, \delta_z^i \in \{-\lambda, 0, \lambda\} \end{cases} \tag{4.21}$$

where $(x_{i+1}, y_{i+1}, z_{i+1})$ is the next path point v_{i+1} choosing to move, and (x_i, y_i, z_i) is the current position v_i. Note that δ_x^i, δ_y^i, and δ_z^i are the movements corresponding to the x-axis, y-axis, and z-axis, respectively. The step size of the movements is λ.

Another constraint is to ensure the planned flight does not collide with buildings. The path of a UAV flight should not enter the airspace units that are occupied by buildings.

$$u_k \in U_{\mathrm{Obs}}, k \geq 0 \tag{4.22}$$

in which u_k is the building-occupied airspace unit, and U_{Obs} is the set of all the building-occupied units. The motivation for making Eq. (4.22) a constraint is to improve computational efficiency. The building-occupied airspace blocks will be disabled during the path-searching process, and they will not be evaluated by the fitness function which is the most time-consuming part of the algorithm.

4.2.3 Solution algorithm for cost-based path planning

This section develops a hybrid algorithm to solve the cost-based path optimization problem. The algorithm incorporates the estimation of distribution algorithm (EDA) and the A* algorithm. To improve computational efficiency, the k-means method is introduced to identify low-cost clusters that are used as local heuristic guidance for path searching. Detailed developments of the algorithm are presented in the following sections.

A. EDA-A algorithm for cost-based path planning*

To solve the developed cost-based path optimization problem, there are several types of algorithms we can choose from. The exact method, such as the Dijkstra, is one of the classic and effective methods for graph-based path-searching. However, Dijkstra is a computational inefficiency method, especially in dealing with large-scale problems such as the one in this Chapter. To improve efficiency, an extension of the Dijkstra method was made with heuristic information, and a new algorithm called the A* algorithm was developed [160]. The computational efficiency of the A* algorithm is significantly higher than the Dijkstra method. While the A* algorithm can still be an exact method if an appropriate heuristic value can be obtained. In the distance-based environment (Fig. 4.6(a)), the heuristic value can be surely obtained either as Manhattan distance or Euclidean distance. However, in the cost-based environment (Fig. 4.6(b)), an appropriate heuristic value is difficult to be obtained because the distribution of cost values is uneven and unpredictable in different airspace units.

If the heuristic value is taken too big, it may exceed the lower bound and that reduces the heuristic effectiveness. In contrast, if the heuristic value is too small, although it might be within the lower bound, the computational efficiency will be significantly reduced. Finding a proper heuristic value can make a good compromise between solution quality and computational efficiency. That motivates this Chapter to propose a global optimization method to assist the estimation of the exact lower bound. What is more, as the scale of the problem gets larger, it becomes more difficult for the heuristic methods that are initiated with only one solution to search for the outstanding solutions from the feasible region.

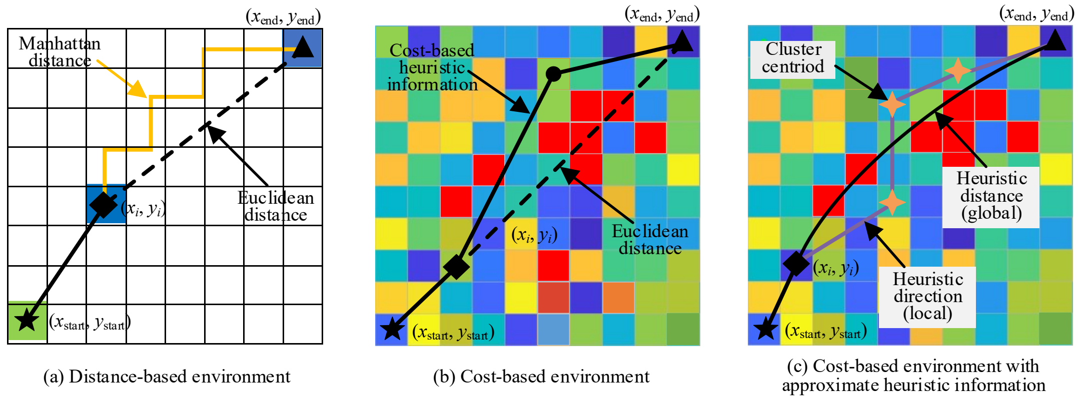


Fig. 4.6: Illustration of heuristic value in the distance-based environment and the cost-based environment. Note that (a) shows that in the distance-based environment, the conventional A* algorithm has two options to choose its heuristic value: Manhattan distance and Euclidean distance. For (b) in the cost-based environment, the heuristic information may not be effective, as the risk cost distribution is uneven and unpredictable from the current node (x_i, y_i) to the target node $(x_{\text{end}}, y_{\text{end}})$. For (c), the global heuristic distance path is directly from the current node to the destination. While the local heuristic direction is generated by connecting the cluster centroids. The heuristic value is approximate to the exact lower bound by incorporating both the global and local heuristic information.

Swarm-based methods initiate with a number of solutions and are suitable to solve the above-proposed problem. The estimation of distribution algorithm (EDA) [161], [162] is one of the typical swarm-based algorithms for solving both discrete and continuous optimization problems. The EDA algorithm is a stochastic method, and it performs well in terms of global searching. That makes it a suitable method to optimize the searching region and estimate the lower bound of the heuristic value in cost-based environments. Besides, the EDA method also has no limitation for the number of variables in the optimization process. That aligns with the requirement for the optimization variable of this Chapter. As in this Chapter, the optimization variables of the cost-based path planning problem are coordinates of path points, and the number of the points may not be the same for different flight paths.

The core of EDA is to generate and sample explicit probabilistic models of promising solutions to guide the search for the optimum. The optimization process can be seen as a series of incremental updates of the probabilistic model to achieve the global optimal solution. To reduce the complexity of the optimization process, authors [163] proposed a method by using a binary integer to avoid exploring irrelevant areas. Inspired by their works, this Chapter develops a hybrid algorithm called EDA-A*, which integrates the EDA and A* algorithms. The outer loop of the EDA-A* algorithm is to solve a binary 0-1 optimization problem, which aims for optimizing the feasible region for path searching. The optimized feasible region will be input in the inner loop, where the A* algorithm is used, to generate the cost-effective path. The workflow and pseudocode of the EDA-A* algorithm is presented in Fig. 4.7 and Algorithm 1.

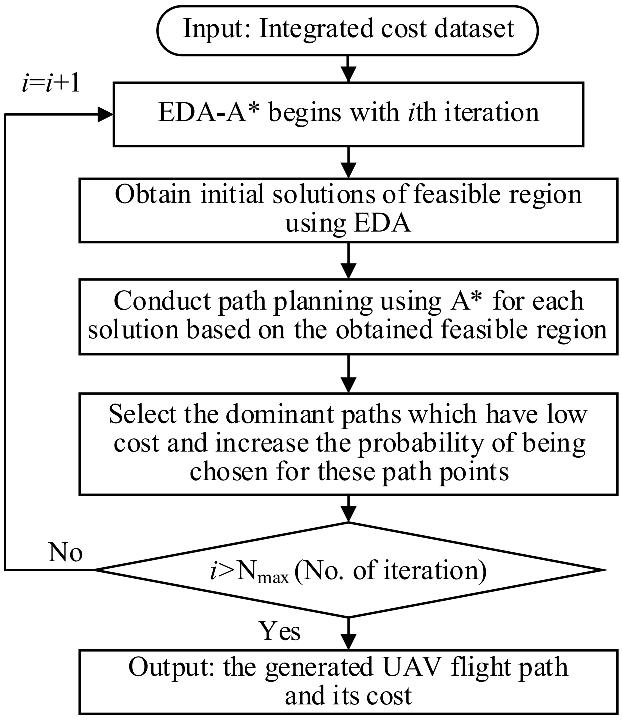


Fig. 4.7: Framework of EDA-A* algorithm.

The main loop of the EDA-A* algorithm is from Line 6 to Line 23 as shown in Algorithm 1. In this algorithm, the species include a number of populations. One population refers to a complete path-searching region based on the selected environment. The EDA method is used to optimize the searching region by selecting the low-cost airspace units along with the reference path from origin to destination. The A* algorithm is used to produce the cost-effective path using the optimized searching region. The essential part of the EDA-A* algorithm is the probability update function (Line 22), which is used to balance the exploitation and exploration of the search process. By selecting the dominant populations from the species, the function is updated towards that individual belonging to the dominant populations will have increased probabilities to be selected as optimal points.

As defined in Eq. (4.23), the probability array $\boldsymbol{p}_{i+1}$ of being selected for next candidate solutions consists of two parts. The first part is to exploit the current best solution by using the currently obtained probability array $\boldsymbol{p}_i$. The second part is to explore the search space for better solutions. Each candidate solution point in the array $\boldsymbol{D}_s$ is divided by the total number of dominant species N_s. The more times a candidate point is selected (based on fitness value), the higher probability it will have in $\boldsymbol{p}_{i+1}$, and eventually it has higher probability to be selected as a solution point. The balance of the exploitation and exploration is adjusted by the learning rate l_{rate}. Specifically, there will be no exploitation process if $l_{\text{rate}} = 1$, as the current best probability has no contribution to the next status. In contrast, there will be no exploration process if $l_{\text{rate}} = 0$, as the next status is completely contributed by the current best probability. The update function is denoted as

$$\boldsymbol{p}_{i+1} = (1 - l_{\text{rate}})\boldsymbol{p}_i + l_{\text{rate}} \frac{\boldsymbol{D}_s}{\mathrm{N}_s} \tag{4.23}$$

Algorithm 1: Hybrid EDA-A* algorithm for cost-based path planning

input: cost value of each airspace unit (Cost), origin and destination (OD) of UAV flight, obstacle information
output: generated path (*path*), total cost of the path (*TotalCost*)
path =*null*;
TotalCost =0;
initialize the probability matrix p of EDA
for $i=1:n_{\text{iter}}$ **Do** % outer loop, n_{iter} is the number of iterations
    while $j<=n_{\text{ppl}}$ **Do** % n_{ppl} is the number of populations in the species
        r=rand(size(Cost)); % initialize the probability (r) of being selected for each airspace unit
        species$\{j,1\}$ = $1.*(r\{j,1\}<p_i\{j,1\})$; % select path searching space by comparing r and p_i
        $j=j+1$;
    end while
    save species % selected path searching spaces
    for $k=1:n_{\text{ppl}}$ **Do** % inner loop
        path =A*(OD, species, obstacle); % generate path using A* algorithm
        TotalCost =FitnessValue(*path*);
        T_cost=[T_cost; *TotalCost*];
    end for
    T_cost(C_np, :)=max(T_cost); % eliminate populations that have no feasible path found
    FitnessValue=T_cost;
    [Fitness, index] = sort(FitnessValue);
    Ds{:, 1} = species{index(:), 1}; % select dominant populations
    p_{i+1}= $(1-l_{\text{rate}})*p_i+l_{\text{rate}}*Ds/\mathrm{N}s$; % probability update function: Eq. (23)
end for

As EDA is a stochastic method based on a probabilistic model, during the iteration, path planning in some populations (feasible regions) might be trapped by obstacles and may not be able to find a feasible path from origin to destination. The total cost $C_{\_\mathrm{np}}$ for these populations will be replaced by the maximum cost among the whole species (Line 18). In subsequent iterations, these populations will be eliminated, and the selected dominant population is guaranteed to find a feasible path.

B. An improvement of EDA-A with fast computation: EDA-CostA**

In EDA-A* algorithm, the EDA is used to select the feasible region for path planning, and A* is used to produce the path based on the selected region. With heuristic information, the A* algorithm conducts very fast compared with the Dijkstra method. However, the A* algorithm is called for every population of the species at every iteration (a for loop from Line 13 to Line 17 in Algorithm 1). That makes the overall EDA-A* algorithm take more computational time as the problem scale gets larger. To cope with this problem, we further improve the EDA-A* algorithm by introducing the k-means method to provide both global and local heuristic information for path searching (see Fig. 4.6(c)). The improved hybrid algorithm is named EDA-CostA*, and its workflow is presented in Fig. 4.8.

The EDA-CostA* algorithm has three main functions. The first function is the EDA algorithm, which is used to globally optimize the feasible region that has low costs among all searching space. The second function is the k-means algorithm, which is used to cluster the low-cost regions and to identify the global heuristic distance and local heuristic direction. The identified heuristic information will be used to estimate the lower bound of the heuristic value that will be ingested into the third function (CostA*) to generate the cost-effective path.

The global heuristic distance h_{Dist} is improved from Euclidean distance to estimate the total cost from the current node to the destination, which is presented as

$$\begin{cases} h_{\mathrm{Dist}} = \hbar_{\mathrm{heuDist}}\sqrt{(x_D - x_i)^2 + (y_D - y_i)^2 + (z_D - z_i)^2} \\ \hbar_{\mathrm{heuDist}} = \min\left\{\left(\dfrac{1}{N_{V_{\mathrm{open}}}} \sum\limits_{v_i \in V_{\mathrm{open}}} c_{v_i}\right), \left(\dfrac{1}{N_{V_{\mathrm{Ctrs}}}} \sum\limits_{v_j \in V_{\mathrm{Ctrs}}} c_{v_j}\right)\right\} \end{cases} \tag{4.24}$$

in which (x_i, y_i, z_i) is the coordination of the current point and (x_D, y_D, z_D) is the destination point, which is used to compute the Euclidean distance. Note that $\hbar_{\mathrm{heuDist}}$

is the heuristic distance factor, which takes the minimum value among the mean cost of all open vertices $\mathrm{V}_{\mathrm{open}}$ and the mean cost of cluster centroid vertices $\mathrm{V}_{\mathrm{Ctrs}}$. Taking the minimum value is because a smaller value is more likely to fall within the lower bound and that leads to a better quality of solution for CostA* algorithm [160]. When the heuristic value is down to zero, the CostA* will then be equivalent to the Dijkstra method. The determination of the heuristic value for the algorithm is to make a trade-off between the quality of the solution and computational efficiency.

The local heuristic direction h_{Drctn} is defined as the distance from the current point (x_i, y_i, z_i) to its nearest cluster centroid $(x_{\mathrm{cen\_}o}, y_{\mathrm{cen\_}o}, z_{\mathrm{cen\_}o})$ as illustrated in Fig. 4.6(c), which is denoted as

$$h_{\mathrm{Drctn}} = \sqrt{(x_{\mathrm{cen\_}o} - x_i)^2 + (y_{\mathrm{cen\_}o} - y_i)^2 + (z_{\mathrm{cen\_}o} - z_i)^2} \tag{4.25}$$

The detailed pseudocode of the EDA-CostA* algorithm is presented as Algorithm 2.

The identified heuristic information above will be ingested into the third function (CostA*) to generate the cost-effective path. In the CostA* algorithm, the solution point is evaluated by the cost function $f(c)$, which is defined as the sum of costs from two parts, as denoted in Eq. (4.26). The first part is the integral of the cost from the origin point to the current point, presented as $g(c)$. The second part is the heuristic value $h(c)$, which is determined by both global heuristic information h_{Dist} and local heuristic information h_{Drctn}.

Algorithm 2: EDA-CostA* algorithm for fast cost-based path planning

input: cost value of each airspace unit (Cost), origin and destination (OD) of UAV flight, obstacle information
output: generated path (*path*), total cost of the path (*TotalCost*)
path =*null*
TotalCost =0;
$\mathrm{P}_{\mathrm{best}}$=EDA(Cost); % obtain the best population $\mathrm{P}_{\mathrm{best}}$
if sum($\mathrm{P}_{\mathrm{best}}(i, j, k)$)==1 **Do** % obtain open points for path searching
    $\mathrm{P}_{\mathrm{indiv}}=[i, j, k]$; % get the position of individual open point
    $\mathrm{P}_{\mathrm{all}}=[\mathrm{P}_{\mathrm{all}}; \mathrm{P}_{\mathrm{indiv}}]$; % obtain the positions of all open points
end if
$\mathrm{P}_{\mathrm{centroid}}$=k-means(Posn); % obtain the positions of cluster centroids
$[h_{\mathrm{Dist}}, h_{\mathrm{Drctn}}]=f(\mathrm{P}_{\mathrm{centroid}})$ % obtain heuristic information by Eq. (24) and Eq. (25)
[*path* , *TotalCost*]=CostA*(OD, obstacle, h_{Dist}, h_{Drctn});

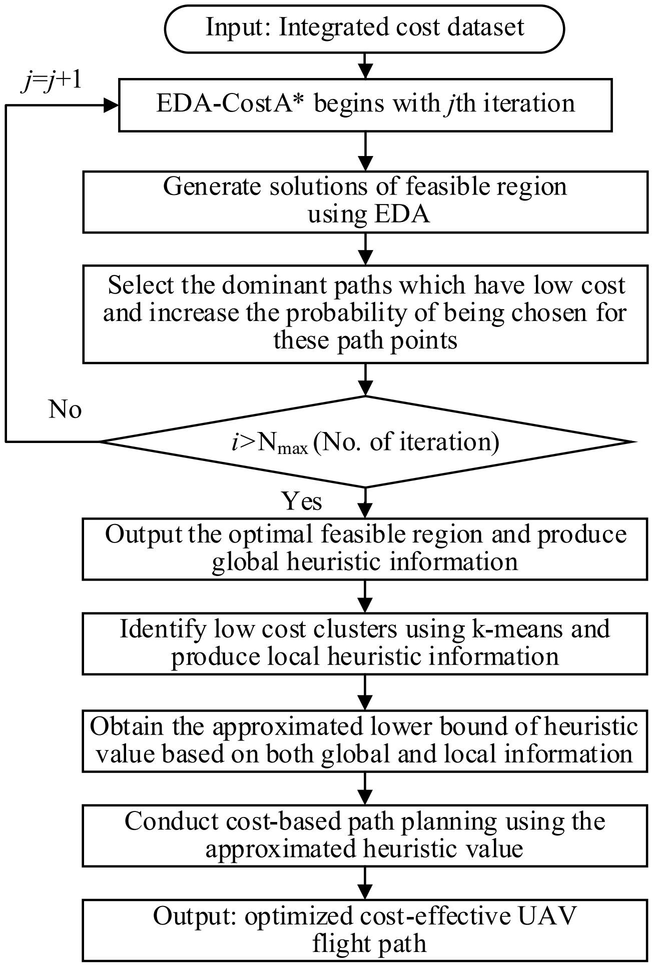


Fig. 4.8: Workflow of EDA-CostA* algorithm.

As Fig. 4.6(c) illustrates, the heuristic distance path provides the global heuristic information h_{Dist}, which is directly from the current node to the destination. While the heuristic direction path provides local heuristic information h_{Drctn}, which connects the cluster centroids. To avoid the local path search deviating too far from the global path, the deviation between the local path and the global path is evaluated and constrained. The deviation is defined as the difference between h_{Drctn} and $h_{\mathrm{Dist\_cen}}$ over $h_{\mathrm{Dist\_cen}}$. Here $h_{\mathrm{Dist\_cen}}$ is the heuristic distance from the current point to the point that is projected from the cluster centroid on the global track. If the deviation does not exceed the threshold ε, which means the local cluster centroid is not far from the global track. The h_{Drctn} will be used as the heuristic information for the current point. In contrast, the h_{Dist} will be used. Here the ε is taken as 0.2, which presents the deviation between the global path and the local path. The cost function is denoted as

$$f(c) = g(c) + h(c) = \int_{v_o}^{v_i} c_v dv + h(c)$$
$$\begin{cases} h(c) = h_{\text{Drctn}}, \; if \; \dfrac{(h_{\text{Drctn}} - h_{\text{Dist\_cen}})}{h_{\text{Dist\_cen}}} < \varepsilon \\ h(c) = h_{\text{Dist}}, \; if \; \dfrac{(h_{\text{Drctn}} - h_{\text{Dist\_cen}})}{h_{\text{Dist\_cen}}} \geq \varepsilon \end{cases} \tag{4.26}$$

The pseudocode of the CostA* algorithm is presented as Algorithm 3.

Algorithm 3: CostA* algorithm

```
input: cost value of each airspace unit (Cost), origin and destination (OD) of UAV flight, obstacle information
output: generated path (path)
for i = 1:26 Do % possible choices of next path point
    next = [x(i), y(i), z(i), Cost(x(i), y(i), z(i))]; % coordinates of the next point and its cost value
    Motion=[Motion; next];
end for
MotionMode{} = Motion; % store the cost value matrix for every open point
g(c)=f(MotionMode{c}); % the integral of the cost from origin point to current point
Obtained the global heuristic information h_Dist by Eq. (24) and local heuristic information h_Drctn by Eq. (25)
if ((h_Drctn - h_Dist_cen)/h_Dist_cen) <ε Do % approximate heuristic value by Eq. (26)
    h(c)=h_Drctn; % deviation is acceptable, path searching direction follows local low-cost cluster
else
    h(c)=h_Dist; % deviation is unacceptable, path searching direction follows global track
end if
f(c)=g(c)+h(c); % overall cost function
if f(c) < open(f(:)) Do % distance from current point to destination less than that of the points in open list
    Putting current point as father point, and it will be included in the path
end if
```

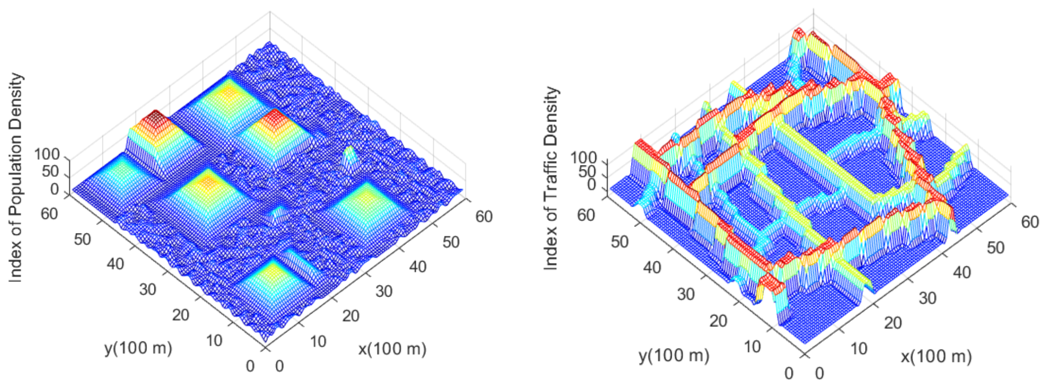
To summarize, there are two similarities between the EDA-A* and EDA-CostA* algorithms. They use the same cost data as input, and they both employ EDA to generate the initial solution of feasible regions. The difference between the two algorithms is the computational efficiency. In EDA-A*, the A* algorithm is called for every single iteration to produce the path for all populations, which consumes significant computational time. While for EDA-CostA*, the CostA* is only called once to generate the cost-effective path based on the optimized feasible region by EDA, which significantly reduces the computational time.

4.3 Simulation Studies and Results

To demonstrate the integrated cost assessment model and the developed cost-based path planning algorithms, we perform simulations and case studies in a representative metropolitan area. First, the cost assessment model is implemented in a real-world environment to generate the cost-based airspace map. Based on the map, the proposed algorithms are used to produce a cost-effective path. Simulations and statistical analysis are conducted to test how well the proposed cost assessment model and algorithms can be generalized to other urban patterns.

4.3.1 Case study of the integrated cost assessment model

A typical metropolitan area (6km×6km) in Singapore is selected for the modelling of risk-based airspace, and the allowable altitude in this study is chosen as 120 meters (400 feet) above the ground. The size of each air block is 100m×100m×30m. The selected metropolitan area has dense high-rise buildings, shopping centers, city squares, residential areas with dense populations, parks, etc., which are representative of modern megacities. A total number of 3366 buildings are included in the selected area. The selected environment has two administrative districts, and the average population densities are 8.358×10^3 and 7.219×10^3 (people/km^2) [164]. The average traffic density in the road of the given area is obtained as 7.12×10^3 (vehicle/km^2) [165]. Based on the average population and vehicle densities, we estimate the population density distribution using Eq. (4.11) and traffic density distribution using Eq. (4.12). Obtained population and traffic distribution results are illustrated in Fig. 4.9.



(a) Population density distribution (b) Traffic density distribution

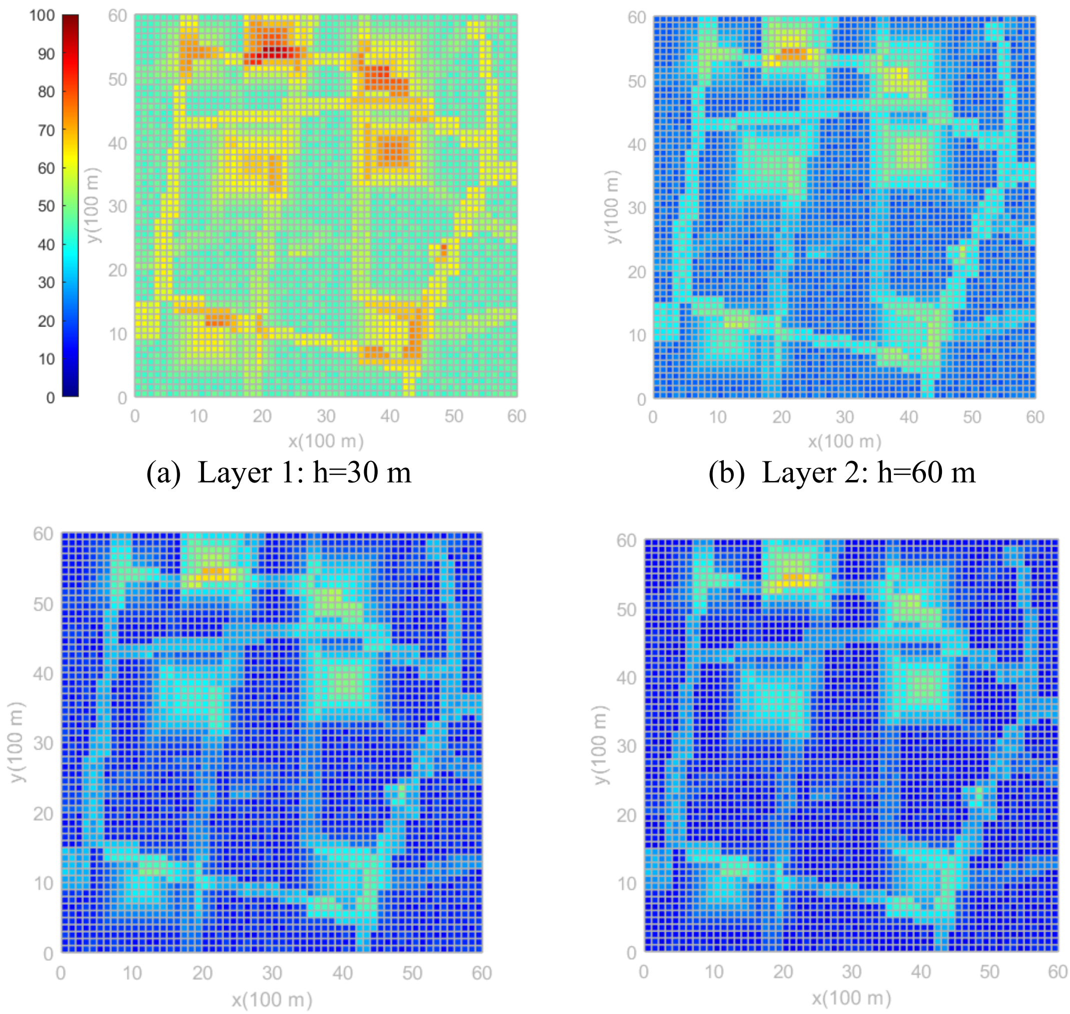
Fig. 4.9: Distribution of population density and traffic density in the selected urban environment.

In this case study, the UAV is selected as one of the most used drones (DJI Phantom 4). The drone weight is 1.38 kg and in urban environments the crash probability P_{crash} is 3.42×10^{-4} per flight hour [8]. The size of the UAV crash impact area is $S_{\_\text{hit}} = 0.0188$ m^2 and the drag coefficient is $R_I = 0.3$ [30]. The number of casualties caused by an average traffic accident is $R_f^{\text{v}} = 0.27$ [166]. For the integrated cost model, the weight factors of fatality risk, property damage risk and noise impact are given as α_1= 0.5, α_2= 0.25, α_3= 0.25, respectively. The fatality risk cost is given a high weightage of 0.5, while the property damage cost and noise impact cost are given the same weight as 0.25 in this study. Based on the obtained data, the total integrated cost of each flight layer is computed and illustrated in Fig. 4.10.

In Fig. 4.10(a), the average cost of flight Layer 1 is the highest. The locations with high fatality risks are identified on the map. For instance, the location (20, 55) on the map has the highest risk cost, as there are shopping streets, highway intersections, and dense populations in real-world environments. Thus, the fatality risk and property damage risk there is high, making the total cost high. In Layer 2 (Fig. 4.10(b)), with the increase of flight altitude, the costs are significantly reduced for property damage and noise impact, whereas the fatality risk cost increases by 7.7% compared with the one in Layer 1. In the third and fourth flight layers, as shown in Fig. 4.10(c) and Fig. 4.10(d), the flight altitude increases to 90 and 120 meters. The high-risk areas in these two layers are still clearly identified while the total cost has not changed much from Layer 3 to Layer 4 because of two reasons. For one thing, the fatality risk only slightly increases by 4.09% from Layer 2 to Layer 3 and 2.66% from Layer 3 to Layer 4 after the altitude is over 60 meters. That is because UAV impact over such height is mostly causing fatalities. For another, the influence of the noise impact exceeds the height threshold (Eq. (4.17) and Fig. 4.5(b)) and contributes nothing to the integrated cost, while the property damage cost is significantly small. The integrated costs in Layer 3 and Layer 4 are therefore significantly small while the high-cost areas are identified.

4.3.2 Cost-based flight path planning

Based on the obtained cost-based map, we conduct the cost-based path planning with algorithm comparisons in terms of solution quality and computational time. The heuristic effectiveness of our proposed EDA-based method is also demonstrated by simulation results. Lastly, the influence of different risk types on the cost-based path planning results is investigated.



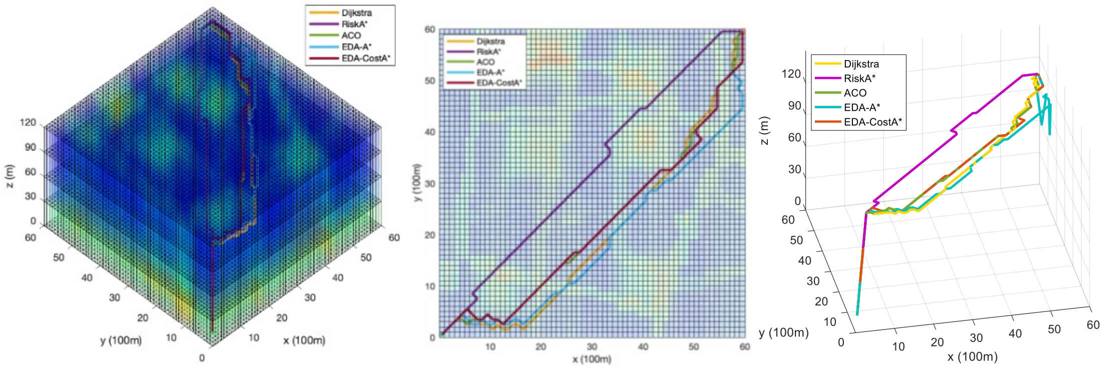
(a) Layer 1: h=30 m
(b) Layer 2: h=60 m
(c) Layer 3: h=90 m
(d) Layer 4: h=120 m

Fig. 4.10: Integrated risk cost mapping for different flight layers.

4.3.2.1. Cost-based path planning in a real-world environment

The flight paths optimized by our proposed hybrid algorithm can avoid obstacles and high-cost areas identified by our proposed cost assessment model. The results of the 3D view with risk map, top view, and 3D view without risk map are presented in Fig. 4.11(a), Fig. 4.11(b), and Fig. 4.11(c). Observed from a 3D view, the drone flight height for most of the time is 120 meters, which is the top layer of the modelled environments. Flying at such height significantly reduced the property damage risk and noise impact cost, and the fatality cost can also be reduced by avoiding high population density and vehicle density areas such as locations (10, 12), (40, 40) and (42, 39) shown in the top view Fig. 4.11(b). In a real-world environment, these identified high-risk cost locations are shopping streets, hospitals, schools, or highway conjunctions, where the population density and vehicle density are significantly higher than the rest of the areas.

Being able to quantitively identify high-risk areas using our proposed model can facilitate the risk management of low-altitude urban airspace and cost-based path optimization. Which subsequently enables safe UAV operations in metropolitan environments.



(a) 3D view (b) Top view (c) 3D view (without risk cost map)

Fig. 4.11: Cost-based paths generated in a real-world environment.

In the simulations, five algorithms are used to generate the cost-based flight path. They are Dijkstra, RiskA*, ACO, EDA-A* and EDA-CostA* algorithms. Dijkstra algorithm, as an exact method, is used to generate the optimal solutions as a basis for comparison. The other two main types of cost-based path planning algorithms are the heuristic A*-based (RiskA*) method [85] and the evolutionary ACO (Ant Colony Optimization: ACO)-based method [22], [167]. In this study, the heuristic value of the RiskA* algorithm is taken as the average cost of all cells in the map. The same cost-based map is used for all five algorithms for path planning.

The path planning results of five algorithms are obtained and presented in Table 4.1. Compared with the Dijkstra algorithm, the flight path produced by EDA-CostA* is 2.06% shorter while it uses a mere 3.05% computational time, whereas the risk cost of the path is 4.58% greater than the Dijkstra one. For the path produced by the ACO algorithm, it has a 1.51% more risk cost and a 1.22% longer distance. However, its computational time is significantly greater than all the other algorithms, with 785.98% greater than the Dijkstra method. For EDA-A* algorithm, its performance makes a good trade-off among the risk cost, flight distance, and computational time. As to RiskA* algorithm, it consumes only 0.16% of the time and the path is 90.65% long compared with the path produced by Dijkstra. However, it has the highest cost index, with 28.8% higher than the basis generated by Dijkstra. As it fails to avoid some high-cost areas such as locations (11, 12) and (43, 45) as shown in Fig. 4.11(b). Furthermore, the heuristic value of RiskA* is also hard to obtain and it plays a key role in optimization performance, as analysis shown in Table 1 of reference [85]. The determination of a proper heuristic value for RiskA* is unsolved, especially when the environment changes. That is the key motivation for our proposed hybrid EDA-A* and EDA-CostA* algorithms to solve the cost-based path planning problems in various environments. Simulations and tests are conducted below to demonstrate the performance of our proposed algorithms.

Table 4.1: Comparison results of the five cost-based path planning algorithms.

Performance\ Indicators	**Computational results**					**Performance ratio compared with Dijkstra (optimal in cost reduction)**			
	Dijkstra	RiskA*	ACO	EDA-A*	EDA-CostA*	RiskA*	ACO	EDA-A*	EDA-CostA*
Risk cost index	1102.6	1420.2	1119.32	1126.26	1153.07	128.80%	101.51%	102.14%	104.58%
Distance (km)	9.73	8.82	9.85	10.11	9.53	90.65%	101.22%	103.94%	97.94%
Computation time (s)	3806.2	6.106	29916.2	984.87	116.22	0.16%	785.98%	25.88%	3.05%

4.3.2.2. Heuristic effectiveness of using EDA to estimate the lower bound

As we discussed in the above Section, it is difficult to obtain the exact heuristic lower bound in a cost-based environment, as the cost values are distributed unevenly and unpredictably. That motivates this Chapter to propose an EDA-based method to better estimate the exact lower bound of heuristic value. As EDA is a stochastic method, the estimated heuristic value may not be a global optimal but suboptimal. We conduct simulations to demonstrate the performance of our proposed EDA-based method in estimating the exact lower bound. The exact lower bound is defined as the exact cost from the current point to the destination, which is the baseline in this study. The baseline value for each path point is obtained by the cost-based Dijkstra method. The estimated lower bound is defined as the approximate cost from the current point to the destination, which is computed by our proposed EDA-based method.

In the simulation, the environment is the same as the one used in Section 4.3.1. A path is generated, and it has 73 path points. Both the exact heuristic value and the estimated heuristic value of each path point are obtained and presented in Fig. 4.12(a).

The absolute error and the relative error between the estimated values and the exact values are also computed, shown in Fig. 4.12(b).

The results in Fig. 4.12(a) show that the estimated heuristic value is estimated to the exact heuristic value, while the estimated values are higher than the exact values except for the values in point 63 and point 71. That enables the high computational efficiency for EDA-CostA* in the path searching process, whereas it may sacrifice the optimality of results as discussed in Table 4.1. The absolute error between the estimated value and exact value shows a decreasing trend (Fig. 4.12(b)), from 93.92 at the origin point to zero at the destination. The relative error shows a stable trend with around 5% error in the first half of the points. The trend intensifies as the points approach the destination. That is because the exact heuristic value (the denominator in the computation of relative error) becomes smaller when approaching the destination. A small change in the difference (the numerator in the computation of relative error) between the estimated value and the exact value will lead to a significant relative error. Overall, the obtained results demonstrate the high heuristic effectiveness of proposed EDA-based methods compared with the baseline, with an average estimation error of 5.65%.

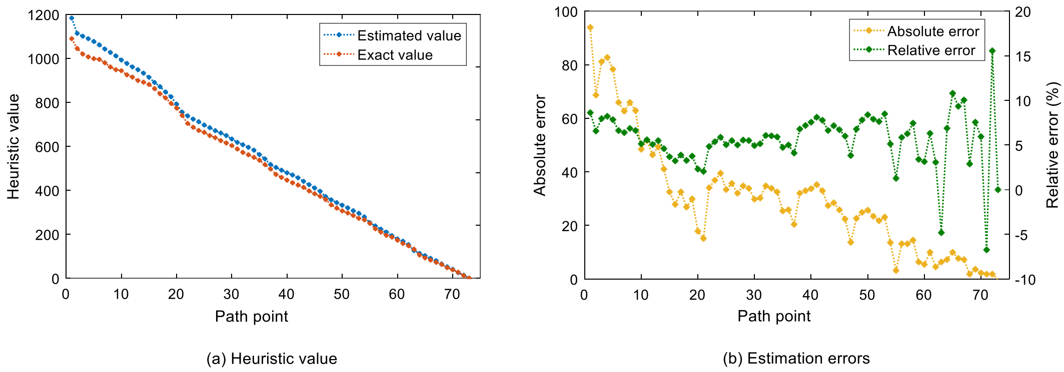


Fig. 4.12: Heuristic effectiveness of our proposed EDA-based methods.

4.3.2.3. *Influence of different risk cost types on cost-based path planning*

To demonstrate the influence of different risk cost types on safe UAV operations, we conduct four groups of path planning simulations: (1) Path1: without considering any risk cost; (2) Path2: only considering fatality risk cost; (3) Path3: considering fatality risk cost and property damage risk cost; (4) Path4: considering all three costs. The environment of the four simulations is the same as the one which is generated in the previous Section. As the EDA-CostA* algorithm performs well among the other methods in terms of computational efficiency and effectiveness, it is applied for each of the four simulations to generate the flight path. Obtained results are shown in Fig. 4.13 and Table 4.2.

Path 1 goes from the origin (1, 1, 1) to the destination (60, 60, 4) with almost a straight line. This path avoids obstacles in locations such as (30, 32) and (40, 40) shown in Fig. 4.13. However, it does not avoid high-cost areas where population density and vehicle density are high, resulting in the total cost of Path1 being the highest among all paths, with a risk cost index of 1698.25 (see Table 4.2). Whereas the distance of the path is the shortest as it goes almost straight to the destination. For Path2 the fatality risk is considered. This path successfully avoids the high population density areas (10, 12), (40, 45), (42, 15) shown in Fig. 4.9(a) and high vehicle density areas (8, 20), (45, 10), and (55, 40) shown in Fig. 4.9(b). As the fatality risk cost has a high proportion in the integrated cost model, avoiding that makes a significant reduction (23.68%) of cost for Path2, compared with Path1. For Path3 the fatality risk and property damage risk are both considered. The produced path3 not only avoids the high fatality risk areas but avoids dense high-rise building areas such as (20, 20) and (30, 30) shown in Fig. 4.4(a), which Path1 and Path2 fail to do so. By adding property damage risk into the model, Path3 can avoid dense building areas, thus the cost of its path further drops by 5.94% compared with Path2. For Path4, it considers all three cost types, the integrated cost of Path4 has a further reduction of 2.48% compared with Path3.

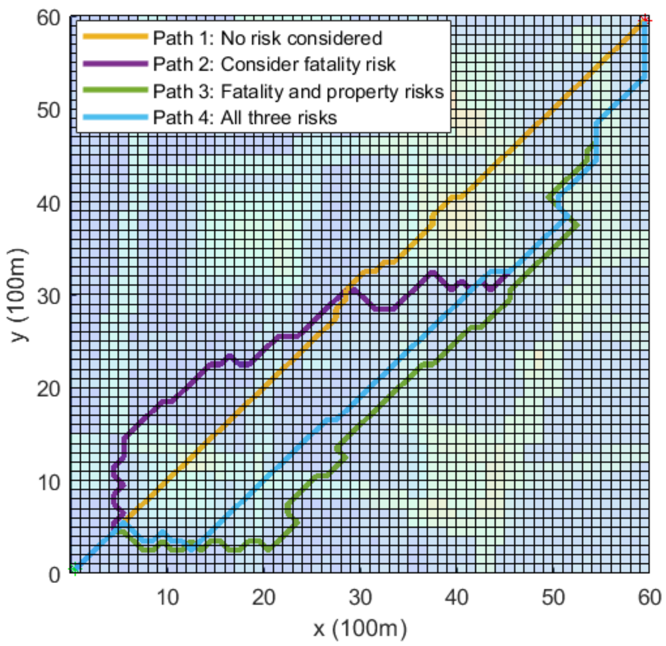


Fig. 4.13: Top view of the generated paths considering different risk cost types.

Overall, the fatality risk cost is the primary factor in the integrated cost assessment model, followed by property damage risk cost and noise impact cost. With more risk types getting considered, a more accurate cost-based map will be generated to produce a lower-cost path for safe UAV operations. While the distance of the produced path will be longer as the UAV may need to travel a longer distance to avoid obstacles and high-risk areas. Analysis of the trade-off between the integrated cost and flight distance is performed in the following subsection.

Table 4.2: Results of the four paths considering different risk cost types.

Performance\ Indicators	**Computational results**				**Performance ratio compared with Path1**			
	Path1	Path2	Path3	Path4	Path1	Path2	Path3	Path4
Risk cost index	1698.25	1296.02	1195.18	1153.10	100.00%	76.32%	70.38%	67.90%
Flight distance (km)	8.35	10.22	10.01	9.53	100.00%	122.32%	119.88%	114.08%

4.3.3 External validity of the integrated cost assessment model

Above, the integrated cost assessment model is used in a specific urban pattern, which is a downtown area selected in Singapore. To demonstrate how well the proposed model can be generalized to other urban patterns, we conduct external validities in this section. Here, external validity refers to how well the outcome of a study can be expected to apply to other settings. In our case, two groups of urban environments are generated for validity. One group has the same size environments, and the other has different size environments. The detailed parameter settings of the environments are given in the following subsections.

4.3.3.1. External validity in environments with the same size

The environmental parameters are randomly generated in the simulations to eliminate the selection bias. Population density is taken from the range of [5, 25] $\times 10^3$ (people/km^2), which covers the most densely populated cities worldwide [168]. The building height distribution of all generated patterns follows a log-normal distribution. The size of the environment is 6km×6km×120m (Length, Width, Height). The integrated cost assessment model is used to compute the cost value of each airspace unit in 100 independent simulation environments, and the distance-based and cost-based Dijkstra methods are used to generate the flight path. The distance-based Dijkstra algorithm is performed in a distance-based map without considering the integrated cost (Eq. (4.20)). While the cost-based Dijkstra is conducted in the cost-based map to minimize the integrated cost. Comparison is made between these two algorithms to see how much percentage of the integrated cost can be reduced by using the cost-based algorithm. The simulation starts from the origin (1, 1, 1) to the destination (60, 60, 4) in the generated environments.

In the 100 sample environments, the total cost is significantly produced by using cost-based Dijkstra. While the flight distance shows the opposite trend. The flight distance of the paths produced by cost-based Dijkstra is greater than that of the distance-based Dijkstra algorithm. To further present the quantitative relationship of the cost and distance, we computed the differences between total cost and total flight distance generated by the above two algorithms for all 100 simulations. The differences are defined as the results obtained by cost-based Dijkstra compared with the results obtained by distance-based Dijkstra. The differences of total cost and flight distance are presented in Fig. 4.14.

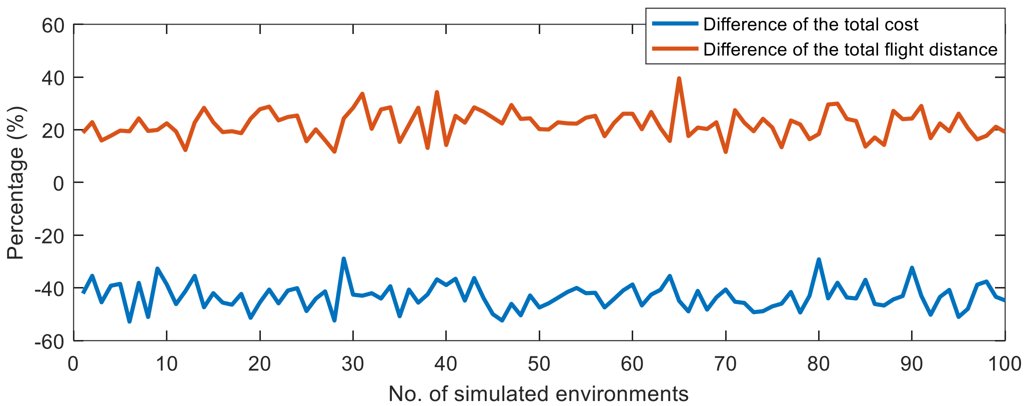


Fig. 4.14: Differences of total cost and total flight distance for the generated path in same size environments (6km×6km×120m). The baseline is the cost and distance of the path produced by the distance-based Dijkstra algorithm. The percentage differences of total cost and total flight distance can be obtained by $(C_{\text{cost\_Dijkstra}} - C_{\text{distance\_Dijkstra}})/C_{\text{distance\_Dijkstra}}$ and $(D_{\text{cost\_Dijkstra}} - D_{\text{distance\_Dijkstra}})/D_{\text{distance\_Dijkstra}}$, respectively.

To test the reliability of the model for the population (all urban patterns), we conducted statistical analysis to find a 95% confidence interval for the cost reduction and flight distance increase. In this case, we have two sets for testing. One is the integrated cost and the other is the flight distance. Each of the tests has two sample groups, and they are the integrated cost of path produced by cost-based Dijkstra (Group 1) and by distance-based Dijkstra (Group 2). Each group has 100 samples. As the sample sizes are large (n_1>30 and n_2>30), we use normal distribution the compute the confidence interval. The sample means ($\bar{x}_1$ and $\bar{x}_2$) and sample variances (s_1^2 and s_2^2) of the two groups are computed and presented in Table 4.3.

The population means of Group 1 and Group 2 are presented as μ_1 and μ_2. The confidence interval of the cost reduction effect can be then described as $(\mu_2 - \mu_1)/\bar{x}_2$, where the interval of $\mu_2 - \mu_1$ can be computed by $(\bar{x}_2 - \bar{x}_1) \pm Z_{\alpha/2}\sqrt{s_1^2/n_1 + s_2^2/n_2}$. The obtained result shows that a 95% confidence interval for cost reduction is $(\mu_2 - \mu_1)/\bar{x}_2 \in [0.4264, 0.4415]$. This means that our proposed integrated cost assessment model is effective for various types of urban patterns, and the average total cost is reduced by [42.64%, 44.15%] at a 95% confidence level. Similarly, we obtained the testing results for flight distance. The average flight distance will increase by [17.36%, 19.23%] at a 95% confidence level by using the cost-based Dijkstra algorithm. That is because UAVs need to travel longer distances to avoid high-cost areas.

Table 4.3: Statistical analysis for cost reduction and flight distance increase.

Groups	Integrated cost (cost index)		Flight distance (m)	
	Cost-based Dijkstra (Group 1)	Distance-based Dijkstra (Group 2)	Cost-based Dijkstra (Group 1)	Distance-based Dijkstra (Group 2)
Sample size	n_1=100	n_2=100	n_1=100	n_2=100
Sample mean	$\bar{x}_1$=18584	$\bar{x}_2$=32831	$\bar{x}_1$=10560	$\bar{x}_2$=8628
Sample variance	s_1^2=1594657	s_2^2=4859067	s_1^2=254418	s_2^2=0

4.3.3.2. External validity in environments with different sizes

Simulations are conducted to further test the reliability of the proposed cost assessment model in environments of different sizes. The simulations include 30 randomly generated environments with the size of 1km×1km, 2km×2km, …, 30km×30km. The height of the environment is 120 meters with four flight layers. The integrated cost assessment model is used to produce the cost-based map, and the cost-based Dijkstra and distance-based Dijkstra algorithms are used to generate the path. The total cost and total distance of the generated path are obtained for both algorithms. The differences between the total cost and total distance by using cost-based Dijkstra, compared with distance-based Dijkstra, are presented in Fig. 4.15.

By using the proposed cost-based path planning method, the average total cost for the 30 environments is reduced by 42.35%, with a variance of 1.54%. Whereas the flight distance increases by 12.03%, with a variance of 0.15%. The results demonstrate that the integrated cost assessment model and cost-based path planning methods are also effective in environments of different sizes. Compared with the results obtained in the same-size (6km×6km) environment, the average total cost increases by [0.29%, 1.80%] while the average total flight distance reduces by [5.33%, 7.23%] in 30 different-size environments. With the increase in environment size, the difference of total cost has no significant change while the difference of total flight distance slightly decreases. The difference of total cost has no significant change because the flight path without considering cost will always go through the high-cost areas that the cost-based path will always avoid (as shown in Section 5.2.2). The increase in environment size only changes the value of total cost but not the percentage of the difference. For the total flight distance, because the search space becomes larger for a flight path with the increase of environment size, which provides more alternatives for the path to avoid high-cost areas. The path can choose the shortest one among the alternatives, which reduces the flight distance.

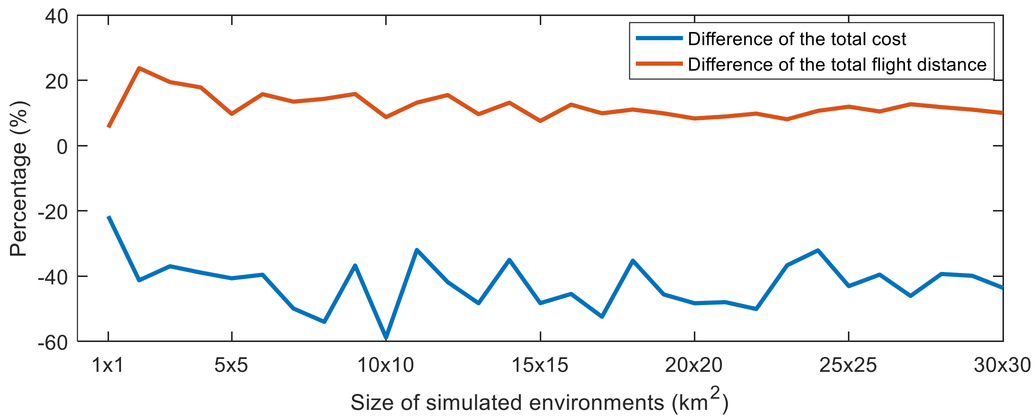


Fig. 4.15: Differences in total cost and total flight distance of the generated path in different sizes of environments. The baseline is the cost and distance of the path produced by the distance-based Dijkstra algorithm.

4.3.4 Reliability analysis of the proposed algorithms

Above we demonstrated that the proposed integrated cost assessment model is effective to generate a cost-based map in various types of urban patterns. With that map, the cost-based algorithm can produce a cost-effective flight path. To test the reliability of proposed cost-based algorithms in different urban environments, we perform simulations based on the urban patterns used in Section 4.3.1. The cost-based Dijkstra algorithm is used to generate the optimal solution as a basis for comparisons with EDA-A* and EDA-CostA* algorithms. Three indicators of the total cost, total flight distance, and computational time are obtained and presented in Fig. 4.16 and Table 4.4.

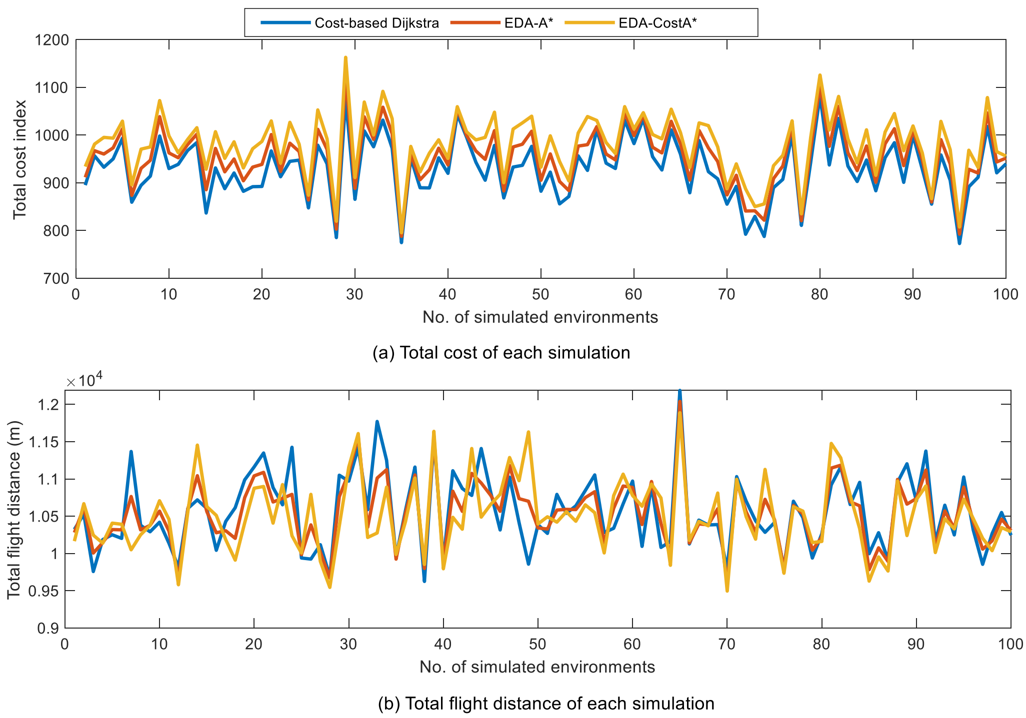


Fig. 4.16: Reliability analysis of the proposed cost-based path planning algorithms.

In the overall 100 simulations, the cost-based Dijkstra algorithm performs the best in terms of total cost reduction, followed by EDA-A* and EDA-CostA* with an average performance rate of 102.76% and 105.47%, respectively. It means that the average total cost of the path produced by EDA-A* and EDA-CostA* algorithms are 2.76% and 5.47% greater than that of the Dijkstra method. On the other hand, the EDA-CostA* algorithm provides the best performance in terms of average total distance and average computational time, with 0.4% shorter total distance and a mere 2.07% in computational time. Followed by EDA-A* algorithm, it saves 0.17% travel distance while spending 24.12% computational time compared with the Dijkstra method. Analysis results show that the proposed EDA-A* and EDA-CostA* algorithms have good robustness, with a mere 1.29% and 2.54% standard deviation in total cost, while 2.18% and 4.29% for total distance (see Table 4.4).

Table 4.4: Reliability analysis results of the proposed algorithms with 100 independent simulations.

Performance\Algorithm	Average total cost		Average total flight distance		Average computation time (s)	Average percentage of the time
	Indicator ratio	Standard Deviation	Indicator ratio	Standard Deviation		
Dijkstra	100.00%	0.00%	100.00%	0.00%	4120.35	100.00%
EDA-A*	102.76%	1.29%	99.83%	2.18%	993.68	24.12%
EDA-CostA*	105.47%	2.54%	99.60%	4.29%	85.24	2.07%

In summary, the EDA-CostA* is the fastest algorithm in terms of computation and it is much faster than the other two algorithms. While it also has a good performance in the quality of solutions. Its performance is made possible by the heuristic information scheme and the algorithm's structure (see Algorithm 3) without the need to call the A* algorithm in every loop. The EDA-CostA* algorithm can be used for large-scale and time-sensitive cost-based path planning applications. The EDA-A* algorithm performs well in both solution quality and computational efficiency, and it can be applied to the trade-off case. Cost-based Dijkstra can be employed to produce the minimum cost path if computational time is not sensitive.

4.4 Summary

Risk mitigation for UAV operation is a key enabler for the opening of airspace to entice investment and promote innovations in this field. By conducting risk cost-based path planning, the UAV operational risks and impacts on populations in urban areas can be significantly reduced before flight. In this Chapter, the UAV flight path optimization problem has been studied with consideration of an integrated cost assessment model of fatality risk, property damage risk, and noise impact for safe UAV operations in metropolitan environments.

First, the developed UAV flight path optimization method is effective and reliable in minimizing the risk cost of the flight path. The method has been applied to a downtown area in Singapore and to 100 sample environments to optimize flight paths. The total integrated costs of the optimized paths have an average reduction by [42.64%, 44.15%] at a 95% confidence level, compared with the costs of the paths produced by the distance-based Dijkstra algorithm. Whereas the average flight distance of the path increased by [17.36%, 19.23%]. That indicates a trade-off exists between the risk cost and flight distance. Because UAVs may need to travel a longer distance to avoid not only obstacles but high-risk cost areas [90].

Second, the proposed integrated cost assessment model can capture comprehensive risk types in urban environments including fatality risk, property damage risk, and noise impact. Instead of taking the average population density into the fatality risk cost model, this Chapter introduced a gravity model to estimate population density distribution on a finer scale. This enables the opening of more airspace, as only the high population density areas will be identified.

Third, the extension of third-party risk in societal indicators (property damage risk and noise impact) is conducive for UAV flight path optimization in cost-based environments. The flight path will tend to fly at a lower altitude if only considers fatality risk, as the fatality risk is smaller at a lower flight altitude due to the kinetic energy being lower. However, UAVs flying at lower altitude encounter more risks of property damage, operational efficiency loss, and noise impact. Because in lower altitude airspace the buildings are denser, which increases the collision probability of UAVs with buildings. That also limits the flight speed and increases the intensity of flight maneuvers, resulting in a loss of operational efficiency [169]. Noise impact also matters, and it can be solved by flying at a higher altitude.

Finally, the developed cost-based path planning algorithm provides a novel scheme to determine the heuristic information for path-searching methods in cost-based environments. The simulation results show the effectiveness and reliability of the developed algorithm in solving cost-based path optimization problems. The proposed hybrid EDA-CostA* algorithm performs best in computational time consuming only 2.07% of the time compared with the Dijkstra method, while it can reach an average of 94% optimality of solution quality.

Chapter 5

5 Stochastic Route Optimization under Ground Risk Uncertainties

Chapter 4 studies the deterministic ground risk assessment and mitigation problems, where population density is considered a static parameter. To handle uncertainties in ground risk studies due to population movement, this Chapter studies the stochastic route optimization model to jointly make flight approval and execution decisions. In the first stage, the approval decision is made by checking the feasibility of flight missions based on the target level of safety and flight duration to eliminate unsafe flight plans. The route selection and departure time decisions are made in the second stage by considering stochastic risk conditions to further reduce the ground risk. The method developed in this Chapter is at the time scale of the pre-tactical flight phase to optimize the flight approval and execution decisions on the day of operations.

5.1 Introduction

The predicted large-scale UAS operations [4] in urban environments bring severe safety risk issues, as unmanned aircraft may crash and cause fatality risk to people on the ground [9], [147]. This poses significant challenges for aviation authorities and UAS operators to safely approve and execute UAS flight missions in urban airspace where people's movement can result in stochastic ground risk conditions. Aviation authorities are required to make the approval decision for a specific UAS flight plan, as they need to ensure that the operational risk of a flight plan is in an acceptable target level of safety (TLOS). TLOS can be determined by referring to the general aviation standards [27]as 10^{-6} (fatalities/flight hour) [27]. However, the evaluation method of fatality risk for a specific flight plan based on its flight route and estimated time of departure (ETD) is still missing, especially under stochastic risk conditions [170]. UAS operators, on the other hand, need to decide the flight plan by optimizing flight routes and ETD under stochastic risk conditions to reduce the operational risk. That is to meet the TLOS requirement and to reduce the operating costs tied to the fatality risks (e.g., insurance fees for flight missions) [166].

The interdependencies of flight approval and execution decisions are complicated due to stochastic risk conditions. As an approved flight plan may not be safe when the time of execution, executing this flight plan may cause unacceptable safety risks (measured by TLOS). While an initially unapproved flight plan could become safe when executed, which yields an efficiency loss for operators. To enhance safety and improve efficiency, an integrated model of flight approval and execution is needed to jointly optimize flight approval and execution (route selection and ETD) decisions. Specifically, a risk assessment model is required to evaluate the fatality risk of a flight plan to support the approval decision. While a risk-based route planning method is needed to generate the risk-minimizing route sets under different risk conditions as input for route selection. Lastly, a stochastic route optimization model is required to make joint decisions under uncertain risk conditions to reduce the total expected fatality risk cost of all flight plans.

Stochastic programming provides a robust framework to handle decision-making problems under uncertainties [171]–[173], which provides a powerful tool to develop the stochastic optimization model in this work. It evolved from deterministic linear programming with the introduction of time-dependent random variables, which allows it to capture the stochasticity of the optimization problem with more robust performance compared with deterministic methods [174]. Thanks to its strengths, stochastic programming has been widely applied in various fields such as supply chain [175], vehicle route problems [176], and solar power management [172]. In these models, the optimization problem was typically formulated as two-stage stochastic programming to approximate a full sequential multistage problem, which has shown a good trade-off between optimization performance and computational cost [171], [176]. In their settings, the first-stage decision was used to make strategic decisions such as route network generation and selection of power storage unit, while the decisions made in the second stage are to tactically optimize the actions in response to observed uncertainties.

In the aviation field, several problems fit into the strengths of stochastic programming, which has joint decision-making capabilities for strategic and tactical planning by different stakeholders [171], [177], [178]. One typical problem is airport scheduling and operations [171]. Strategic airport scheduling is to control the overcapacity of arranged flights being cancelled due to limited airport capacity, which is determined a few months before operations by the airport side. Tactical interventions for flight operation based on observed uncertainties (e.g., weather) help to reduce delay costs, which are decided a few hours or minutes before departure by the air traffic controller. Another branch of air traffic management studies handles flight arrival decisions of sequencing and scheduling with two-stage stochastic optimization models [177], [178]. In these settings, the scheduling is considered as the first-stage decision that is under uncertain traffic conditions at arrival points, and the decision variable at this stage is taken as the time of arrival at a sequencing point. In the second stage, the sequencing decision is made with a binary variable to determine the order of arriving flights. Their results demonstrated that the two-stage stochastic model performs well in resolving joint decision-making problems.

After the discussions above, one key research gap is that existing works treated the ground risk assessment and mitigation in a static and separate manner, which fails to provide support for time-dependent route optimization. Another critical research gap is that studies on time-dependent route optimization under stochastic risk conditions have not been seen to jointly optimize flight approval and execution decisions for emerging UAS operations. In this Chapter, an emerging UAS route optimization problem is considered where safety is a paramount factor that requires civil aviation authorities (CAAs) to make the flight approval decision at an acceptable target level of safety. An approval decision is made based on the ground risk of the flight plan, while the risk of the plan is determined by the flight route generated in the risk condition at the time of operation. The more effective the risk-minimizing route, the more flight plans will be approved for execution. These interdependencies are further complicated by risk uncertainties due to population movement. An approved flight mission may violate the TLOS requirement when operating due to ground risk uncertainty, resulting in a safety accident. In contrast, an unapproved flight plan which violated TLOS when made approval decision may become feasible at the time of operating thanks to risk conditions changed. These interdependent decision-making difficulties give rise to the most notable research gap, a lack of an integrated model to jointly optimize flight approval and execution decisions with the objective of minimizing the total ground risk of all UAS flight operations.

This Chapter develops a two-stage stochastic route optimization model to jointly make flight approval and execution decisions with the objective of minimizing the total ground risk of UAS operations under uncertain risk conditions. This Chapter makes four major contributions to the safety risk management of UAS operations.

(i) We propose a time-dependent risk assessment model to evaluate the fatality risk based on stochastic population density for specific UAS flight plans. The model provides support for aviation authorities to decide if a specific flight plan (with flight route and ETD) can be approved under a given target level of safety (TLOS).

(ii) We build a data-driven population density prediction model based on a spatial-temporal public transportation dataset to estimate the population density, which acts as a critical parameter in the time-dependent risk assessment model.

(iii) We propose a discretized time-dependent risk-based route planning algorithm (TD-RiskA*) with a parallel computing scheme. The algorithm enables fast generation of the risk-minimizing route set for all flight OD pairs in polynomial time.

(iv) We formulate the time-dependent risk-based route optimization (TD-RRO) problem as a two-stage stochastic optimization model. The first-stage decision is to determine the feasibility of a flight mission in terms of TLOS and flight duration requirements. While route selection and departure time-related decisions are made in the second stage with the objective of minimizing total ground risk cost. The developed model allows UAS users (delivery or air taxi companies) to optimize the routes and timetables of their flight missions to avoid high-risk conditions.

The rest of this Chapter is organized as follows: Section 5.2 presents the problem statement and definitions of key parameters used in this Chapter. The formulations of the risk assessment model, population density prediction model, and time-dependent risk-based route planning model are conducted in Section 5.3. Section 5.4 develops the time-dependent route optimization model in deterministic and stochastic forms. Numerical analysis and discussions are performed in Section 5.5, followed by the summary in Section 5.6.

5.2 Problem Statement

The aim of this Chapter is to develop a stochastic route optimization model to reduce the ground risk of time-dependent UAS operations over people on the ground, thus ensuring appropriately safe UAS operations that meet the target level of safety (TLOS). In this section, the concepts of time-dependent ground fatality risk and TLOS are introduced, followed by a risk mitigation approach via time-dependent risk-based route planning.

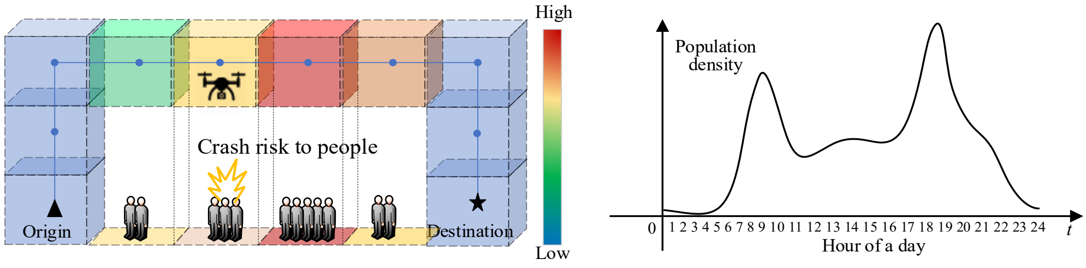
5.2.1 Time-Dependent Ground Risk and Target Level of Safety

Unmanned aircraft may lose control and crash into people on the ground, causing fatalities. To evaluate the ground fatalities caused, a concept of ground risk is proposed, which is defined as the expected number of fatalities caused by UAS operations per flight hour [27]. To quantify the ground risk, the fatality rate $\lambda_{\text{fatality}}$ has been introduced as

$$\lambda_{\text{fatality}} = \lambda_{\text{LOC}} N_{\text{people}} P_{(\text{fata}|\text{coli,LOC})} \tag{5.1}$$

where $\lambda_{\text{fatality}}$ is the expected ground fatality rate per flight hour for a specific UA operation, and λ_{LOC} is the rate of loss of control (LOC) for a specific type of UA, which is determined by the product reliability of the UA and operational failures. Note that N_{people} is the expected number of people being collided during a LOC event, and $P_{(\text{fata}|\text{coli,LOC})}$ is the conditional probability of a UA causing a fatality for the impacted people on condition of a failed UA colliding with that person.

In the above parameters, λ_{LOC} and $P_{(\text{fata}|\text{coli,LOC})}$ are determined variables with a given UA type and operational environment. While N_{people} is a time-dependent variable, which is correlated with the spatial-temporal population density. The population density varies in different spatial locations of a flight route, which subsequently results in different ground risk costs, shown in Fig. 5.1(a). Due to the population movement, the population density at the same location is stochastically distributed, as presented in Fig. 5.1(b). That makes the ground risk a probabilistic time-dependent parameter.



(a) Ground fatality risk associated with population density

(b) Stochastic population density at a particular location

Fig. 5.1: Ground fatality risk associated with spatial-temporal population density.

The ground risk of flying over a grid cell v_i at time t is denoted as $c_{v_i\_t}$, which is equivalent to the ground fatality rate $c_{v_i\_t} \sim \lambda_{v_i\_t}$. The total ground risk cost $C_{P_k\_t}$ of a flight route P_k at time t is the summation of the ground risk of each route grid cell, denoted as

$$C_{P_k\_t} = \sum_{v_i \in P_k} c_{v_i\_t} \tag{5.2}$$

To evaluate if a UA operation is appropriately safe, a quantitative and justifiable TLOS has been defined for UAS operations based on general aviation third-party ground risk [27], denoted as

$$\lambda_{\mathrm{TLOS}} = \lambda_{(\mathrm{GA_{accident}})} N_{(\mathrm{fatal|GA_{accident}})} \tag{5.3}$$

where$\lambda_{(\mathrm{GA_{accident}})}$ is the accepted accident rate for manned general aviation operations. While $N_{(\mathrm{fatal|GA_{accident}})}$ is the expected number of ground fatalities per accident.

With the total ground risk cost of a flight route and the defined TLOS, we can define the condition of a TLOS violation as follows.

Definition 1 (TLOS violation): Let P_k be the route of flight k and let $C_{P_k\_t}$ be the total ground risk cost of the route P_k at time t. The required target level of safety (TLOS) is denoted as λ_{TLOS}. The condition of a TLOS violation is defined as the expected ground risk cost of a flight route is larger than the required TLOS, denoted as

$$C_{P_k\_t} > \lambda_{\mathrm{TLOS}}, P_k \in \mathbb{P}, t \in T \tag{5.4}$$

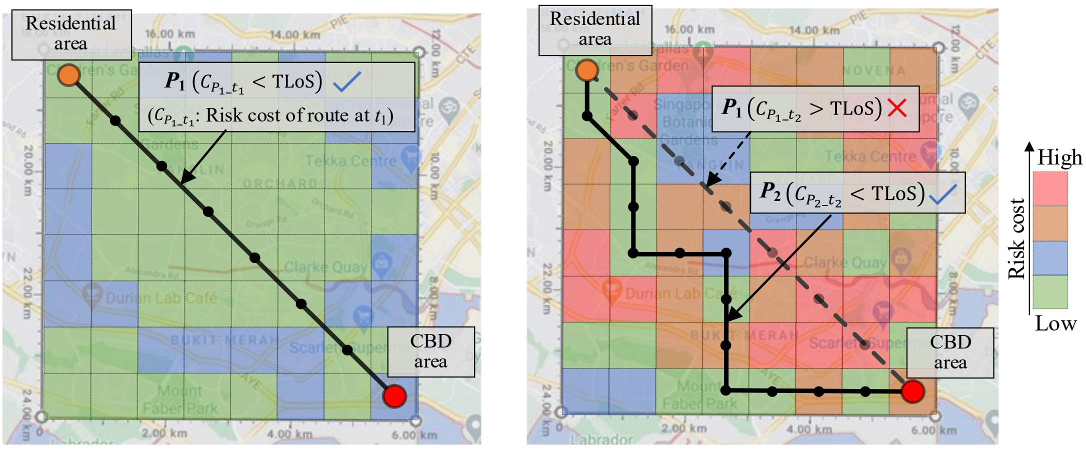
where $\mathbb{P}$ is the set of all types of UA flights and T is the set of all times of the day.

UA operations in urban environments may need to fly over areas with high population density during rush hours (e.g., 7:00-9:00 and 18:00-20:00, as shown in Fig. 5.1(b)) when transport demands are at peaks. That may fail UA operations due to TLOS violations. Reducing the ground risk for UAS operations through appropriate mitigation measures is therefore critically needed.

5.2.2 Risk Mitigation through Risk-Based Route Planning

According to Eq. (5.1), ground risk can be reduced from three aspects: (i) reducing the loss of control rate of UA (i.e., improving the reliability of UA products), (ii) reducing the number of people at risk of UA operations, (iii) reducing the effects of UA impact to people (e.g., using lighter and softer materials). Aspects (i) and (iii) are related to the UA design and manufacturing perspectives, which are out of the scope of this research.

The focus of this Chapter is on aspect (ii) reducing the number of people at risk of UA operations, which is to conduct time-dependent risk-based route optimization to avoid high-risk areas where population densities are high at the time of UA operations. Here, the risk-based route is defined as a flight route with the objective of minimizing the ground risk associated with the waypoints of the route. A feasible risk-based route is defined as a route whose total ground risk is smaller than the required TLOS. An illustration of a feasible route P_1 at time t_1 is shown in Fig. 5.2(a). The total risk cost of this route $C_{P_1\_t_1}$ is less than the required TLOS, making it a feasible route in terms of ground risk. However, when the operation timing changed, for instance, to t_2, the ground risk distribution is changed due to population movement, as illustrated in Fig. 5.2(b). Consequently, the risk cost of the route changed to $C_{P_1\_t_2}$, which is no longer smaller than the TLOS, making it an infeasible route. In the new risk condition (at t_2), another risk-based route P_2 is generated, which is feasible in terms of ground risk requirement ($C_{P_2\_t_2} < \mathrm{TLOS}$). Hence, we introduce the concept of time-dependent risk-based (TDRB) route, which is defined as a set of feasible routes for an OD flight k denoted as $P_k^{\mathrm{TDRB}} = \{P_1, P_2, \ldots, P_m\}, k, m \geq 1;\ k, m \in N_+$. The feasible routes are generated in different timings when the risk conditions are different due to the stochastic population density.



(a) Risk-based flight route at t_1 (5:00 am) (b) Risk-based flight route at t_2 (8:00 am)

Fig. 5.2: Time-dependent risk-based flight routes from a residential area to a central business district (CBD) area in Singapore. Note that a safe route (measured by TLOS) for a given OD pair can be different because of stochastic ground risk conditions due to population movement.

The aim of this Chapter is to minimize the total expected ground risk for UAS operations in the presence of time-dependent fatality risk induced by population movement. While ensuring that all flight routes are feasible in terms of the TLOS requirement. A general form of the time-dependent route optimization problem is formulated as

$$\min\colon \mathbb{E}\left\{C_{P_{k}\_t} \middle| C_{P_{k}\_t} \leq \lambda_{\mathrm{TLOS}}, P_k \in \mathbb{P}, t \in T\right\} \tag{5.5}$$

5.3 Principles of Risk Assessment and Risk-Based Route Planning

In this section, we define the fundamentals of a ground risk assessment model with a core population density parameter. A regression model is developed to predict the population density distribution. With the estimated time-dependent population density, a dynamic risk map is obtained. Based on the risk map, a time-dependent risk-based route planning method is proposed to generate the risk-minimizing routes for each flight plan. This section prepares the formulation of the stochastic route optimization problem in the following section.

5.3.1 Ground Risk Assessment Model

As defined in Eq. (5.1), the ground risk is quantitatively captured by the ground fatality rate $\lambda_{\text{fatality}}$. In this Chapter, our aim is to evaluate the total ground risk of a flight route in grid environments, as defined in Eq. (5.2). The total ground risk is defined as $C_{P_{k\_t}}$, and its basic element is the risk cost $c_{v_{i\_t}}$, which presents the expected casualties per operation per flight hour at the location v_i at time t. Thus, it can be calculated with the fatality rate $\lambda_{v_{i\_t}}$ products the overfly time Δt_{v_i} on the route segment represented by its centroid v_i. The risk cost $c_{v_{i\_t}}$ is denoted as

$$c_{v_{i\_t}} = \lambda_{v_{i\_t}} \Delta t_{v_i} \tag{5.6}$$

$$\lambda_{v_{i\_t}} = \lambda_{\text{LOC}} N_{v_{i\_t}}^{\text{crash}} P_{\text{fatal}} \tag{5.7}$$

where λ_{LOC} is the loss of control rate of a UA, and $N_{v_{i\_t}}^{\text{crash}}$ is the expected number of people that are at risk of a UA crash event at time flight time t, and P_{fatal} is the fatality probability on the condition of a crashed UA hitting a person.

We further decompose $N_{v_{i\_t}}^{\text{crash}}$ into A_{crash} (size of the crash area of a specific UA), S_c (sheltering coefficient), and $d_{v_{i\_t}}$ (population density at a location v_i at time t), denoted as

$$N_{v_{i\_t}}^{\text{crash}} = A_{\text{crash}} S_c d_{v_{i\_t}} \tag{5.8}$$

Here A_{crash} is the function of UA wingspan and flight velocity, and S_c is the fraction of the population that is exposed to the risk of UA operations. Note that only the population density $d_{v_{i\_t}}$ is a time-dependent variable due to people movement over a day in urban environments.

To obtain the population density distribution, studies have been conducted by utilizing two major sources of data. One group uses cellphone data, which gives the exact locations of population distribution at different times [155]. However, cellphone data is highly restricted due to public privacy and security issues. Obtaining a complete cellphone dataset for all populations is also challenging, as people may use different telecommunication networks while some may not allow the cellphone location open while using [179]. The other group of research works [180], [181] uses public transportation data (accounting for 70% of passengers [182]) to capture the patterns of passenger flow and population movement within a city environment, which inspires the prediction of the spatial-temporal population movement in this research.

5.3.2 Population Density Prediction Model

In this Chapter, we utilize the public transportation data to estimate and predict the population density and subsequently derive the time-dependent ground risk for risk-based UA flight route optimization. The definition of population density based on the public transportation data in grid environments is proposed. As shown in Fig. 5.3(a), people moving in and out of train/bus stations are exposed to the risk of UA operations. By looking at the tape-in and tape-out data, the number of people in a location can be obtained, and the population density of this location can be derived from the size of this location. A formal definition of population density is presented as follows.

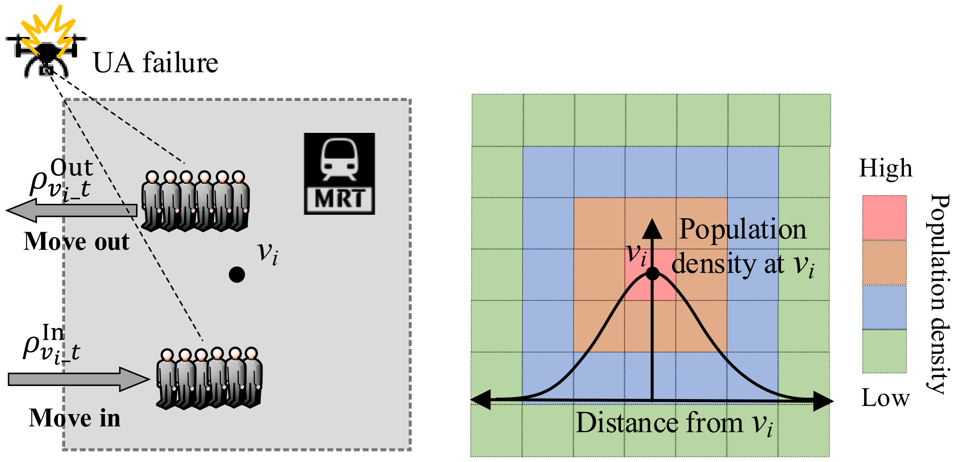
Definition 2 (Population density). Let v_i be the centroid of a grid that a flight travels and let A_{v_i} be the size of this grid. Let $\rho_{v_i\_t}^{\mathrm{In}}$ and $\rho_{v_i\_t}^{\mathrm{Out}}$ be the number of people moving in and moving out of the grid during the time period preceding time t. The population density of the grid v_i at time t can then be defined as

$$d_{v_i\_t} = \left(\frac{\rho_{v_i\_t}^{\mathrm{In}} + \rho_{v_i\_t}^{\mathrm{Out}}}{A_{v_i}}\right) \tag{5.9}$$

The population density for grids that have public transport stations can be obtained by the model above. While for grids without such stations, the diffusion process is needed to approximate a population density, which commonly employs gravity-based models [170]. In the gravity-based model, with the increase in the distance between the grid v_i and its nearby grid $v_{(i+\tau)}$, the population density reduces over the grided map (as illustrated in Fig. 5.3(b)). The grid v_i with stations exerts influence for the nearby grids within its influential distance r with an exponential gradient [52], defined as

$$d_{v_{(i+\tau)}\_t} = \begin{cases} d_{v_i\_t}\left(\dfrac{1}{e^{2r^2}}\right), 0 < r \leq 1 \text{ and } d_{v_i\_t}\left(\dfrac{1}{e^{2r^2}}\right) \geq d_{v_{(i+\tau)}}^{\mathrm{census}} \\ d_{v_{(i+\tau)}}^{\mathrm{census}}, r > 1 \text{ or } d_{v_i\_t}\left(\dfrac{1}{e^{2r^2}}\right) < d_{v_{(i+\tau)}}^{\mathrm{census}} \end{cases} \tag{5.10}$$

where $d_{v_{(i+\tau)}\_t}$ is the estimated population density of the grid $v_{(i+\tau)}$, and $d_{v_{(i+\tau)}}^{\mathrm{census}}$ is the census population density. The diffusion process ceases when the distance between grids exceeds the influential range r. For the rest of the grids that exceed r and have no stations, the population density for them is assigned as the census population density [22], [27].



(a) Population density at v_i (b) Pop. density diffusion from v_i

Fig. 5.3: Conceptual view of the calculation and diffusion process of population density.

In this section, a regression model is used to predict the variable $d_{v_i\_t}$. We assume $d_{v_i\_t}$ follows a normal distribution $d_{v_i\_t} \in N(\mu_{v_i\_t}, \sigma_{v_i\_t}^2)$, with the population mean and variance of $\mu_{v_i\_t}$ and $\sigma_{v_i\_t}^2$, respectively. To estimate the population mean and variance, the sample mean $\hat{d}_{v_i\_t}$ and sample variance $\hat{s}_{v_i\_t}^2$ are introduced, which are derived from the historical observations of the population density. The observations were made at the same location and time. The same time t refers to the time during any day; and observations are made on different days. We define the ρth observation at the location v_i at time t is $d_{v_i\_t}^{\rho} \in \mathbb{R}$. The estimation parameters are denoted as

$$\mu_{v_i\_t} = \hat{d}_{v_i\_t} = \frac{1}{n}\left(\sum_{\rho=1}^{n} d_{v_i\_t}^{\rho}\right), d_{v_i\_t}^{\rho} \in \mathbb{R}, n \in \mathbb{Z}_{+} \tag{5.11}$$

$$\sigma_{v_i\_t}^2 = \hat{s}_{v_i\_t}^2 = \frac{1}{(n-1)}\left(\sum_{\rho=1}^{n} \left(d_{v_i\_t}^{\rho} - \mu_{v_i\_t}\right)^2\right), d_{v_i\_t}^{\rho} \in \mathbb{R}, n \in \mathbb{Z}_{+} \tag{5.12}$$

Note that Eq. (5.11) and Eq. (5.12) define the sample mean and variance derived from the historical observations. Where n is the sample size, which is required to be $n \geq 30$ based on the Central Limited Theorem [183] to justify the normality assumption for the estimation of the population parameters. Note that the sample variance is divided by $(n-1)$ to yield statistically unbiased estimators.

To facilitate the generation of prediction interval, we convert the normal distribution of that variable $d_{v_{i\_}t}$ follows into its standard form, denoted as

$$Z = \frac{\left(d_{v_{i\_}t} - \mu_{v_{i\_}t}\right)}{\sqrt{\sigma_{v_{i\_}t}^2 + \frac{\sigma_{v_{i\_}t}^2}{n}}} = \frac{\left(d_{v_{i\_}t} - \mu_{v_{i\_}t}\right)}{\sigma_{v_{i\_}t}\sqrt{1+\frac{1}{n}}} \sim N(Z;0,1) \tag{5.13}$$

Based on the standard normal distribution, the prediction interval for future observation of $d_{v_{i\_}t}$ under a confidence level of $(1-\alpha)$ is determined as

$$\mathcal{P}\left(-Z_{\frac{\alpha}{2}} < \frac{\left(d_{v_{i\_}t} - \mu_{v_{i\_}t}\right)}{\sigma_{v_{i\_}t}\sqrt{1+\frac{1}{n}}} < Z_{\frac{\alpha}{2}}\right) = (1-\alpha) \tag{5.14}$$

$$\left(\mu_{v_{i\_}t} - Z_{\frac{\alpha}{2}}\sqrt{\frac{\sigma_{v_{i\_}t}^2}{n}}\right) < d_{v_{i\_}t}(\alpha) < \left(\mu_{v_{i\_}t} + Z_{\frac{\alpha}{2}}\sqrt{\frac{\sigma_{v_{i\_}t}^2}{n}}\right) \tag{5.15}$$

In this Chapter, the predicted population density is used to compute the ground risk cost. To ensure the safety of UA operations, we take the conservative prediction value, which is the upper bound of the prediction interval as the future observation of $d_{v_{i\_}t}$, denoted as

$$d_{v_{i\_}t}(\alpha) = \left(\mu_{v_{i\_}t} + Z_{\frac{\alpha}{2}}\sqrt{\frac{\sigma_{v_{i\_}t}^2}{n}}\right) \tag{5.16}$$

An example is given to demonstrate the derivation of the prediction model from historical transportation data. One train interchange station in Singapore is selected with historical passengers' tape-in and tape-out data. The data is taken as hourly weekday average number of passengers, and 52 weeks (n=52) of data is collected, which are shown as grey lines in Fig. 5.4. By using Eqs. (5.11)-(5.15), the prediction lower and upper bounds are obtained as shown in red and blue lines. For a given grid cell v_i, the population density $d_{v_{i\_}t}$ is the function of time $\mathcal{F}(t)$ that a flight flying over that grid cell.

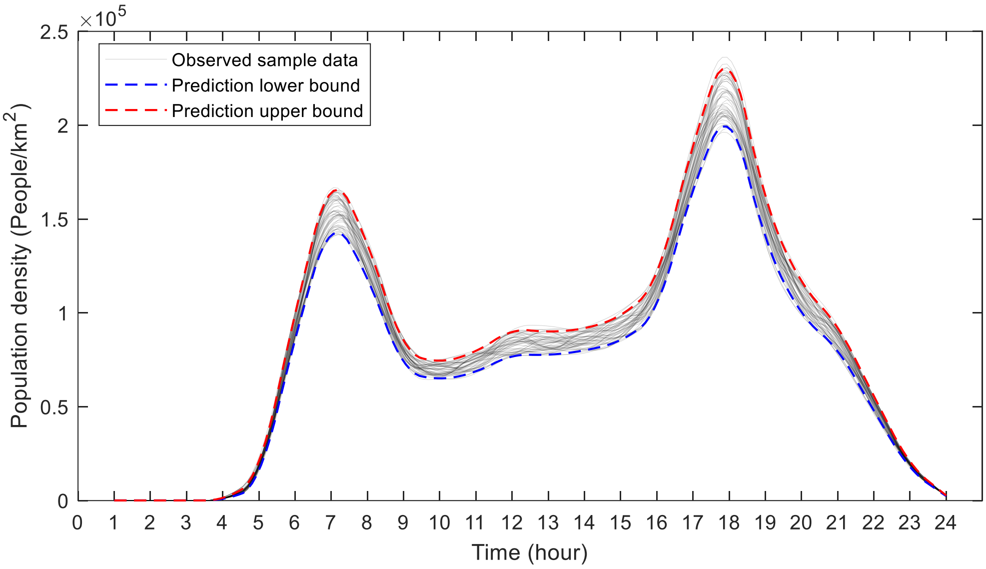


Fig. 5.4: Time-varying prediction interval (at 95% confidence level) of the population density at a typical train interchange station in Singapore.

With the population density prediction model, the time-dependent risk cost maps can be obtained. Based on the maps, time-dependent risk-based route planning can be conducted to find risk-minimizing routes for safe UAS operations.

5.3.3 Time-Dependent Risk-based Route Planning Approach

The time-dependent risk-based route planning problem is to find risk-minimizing routes (measured by ground risk) for an OD pair in risk-based maps under stochastic risk conditions. This problem is considered NP-hard and difficult to solve [184]–[186]. While it becomes tractable if a particular departure time is given, which makes the subproblem a discretized one. To solve the risk-minimizing route planning subproblem at each discretized time point, this Chapter proposes a time-dependent risk-based A* (TD-RiskA*) algorithm, and its detailed developments are shown in Algorithm 1.

The proposed TD-RiskA* algorithm differs from the conventional distance-based A* algorithm. It requires the input of the risk cost map M_t_m at time point t_m. The map is then used to evaluate the risk cost of candidate choices for each route point. The candidate choice is defined as the degree of freedom (DoF) of UAS motions in a grid environment, and the DoF is 26 in 3D space with a motion angle interval of 45 degrees. To evaluate the overall risk cost of each motion from the current point, two functions are needed. One function $g(c)$ is needed to calculate the integral of risk cost from the origin to the current route point. While the other function $h(x)$ is employed to estimate the heuristic value from the current point to the destination, and a well-developed hybrid EDA-(k-means) algorithm [170] is used to estimate the heuristic value in this work. The obtained overall risk cost function $f(x)$ is used in the route search process by evaluating if the current point is the best among the 26 candidate choices of the father point. After searching, the algorithm outputs the risk-minimizing route for a given OD pair at the given time-dependent risk map.

Algorithm 1: Time-dependent risk-based A* (TD-RiskA*) algorithm for risk-minimizing routes generation

i. Initilization with data input
OD pair f_k: map.start=[x(0), y(0), z(0)], map.goal=[x(g), y(g), z(g)]
Time-dependent risk cost map $\boldsymbol{M}\_t_m$: An array stores the risk cost of every grid cell
ii. Evaluataion of risk cost for adjacent points
for j = 1: DoF **Do** % possible choices of next route point, DoF is 26 in 3D space with angle interval of 45 degrees
    next = [$x(j)$, $y(j)$, $z(j)$, $\boldsymbol{M}\_t_m$($x(j)$, $y(j)$, $z(j)$)]; % coordinates of the next point and its risk cost value
    Motion=[Motion; next];
end for
MotionMode{} = Motion; % store the cost value matrix for every open point
iii. Overall risk cost function
g(c)=f(MotionMode{c}); % Calculation of integral of the risk cost from origin to current point
h(c)=EDA-k_means(map, $\boldsymbol{M}\_t_m$); %Estimation of hueristic cost from current point to the destination
f(c)=g(c)+h(c);
IV. Route search process
if f(c) < open(f (:)) **Do** % The risk cost from current point to destination less than that of the points in open list
    Putting current point as father point, and it will be included in the planned route
end if
Output: Route of flight f_k at time t_m (P_f_k_t_m)

The number of executions of each script in the route search loop is n (number of DoF), which makes the complexity of the TD-RiskA* algorithm $O(n)$. However, the generation of risk-minimizing route sets using TD-RiskA* at all discretized time points (m) for all OD pairs (k) requires a more significant number of computations, making the overall complexity of generating route sets $O(n^3)$. To improve computational efficiency, a parallel computing scheme is introduced and presented in Fig. 5.5.

In the parallel computing process, individual flight pair is assigned to every single one of the risk maps associated with different time points to generate the risk-minimizing routes concurrently. By doing so, the computation complexity reduces from $O(n^3)$ to $O(n)$, which enables the fast decision-making of route set generation. The obtained set of routes is considered as candidate options for each OD pair to select in the optimization model.

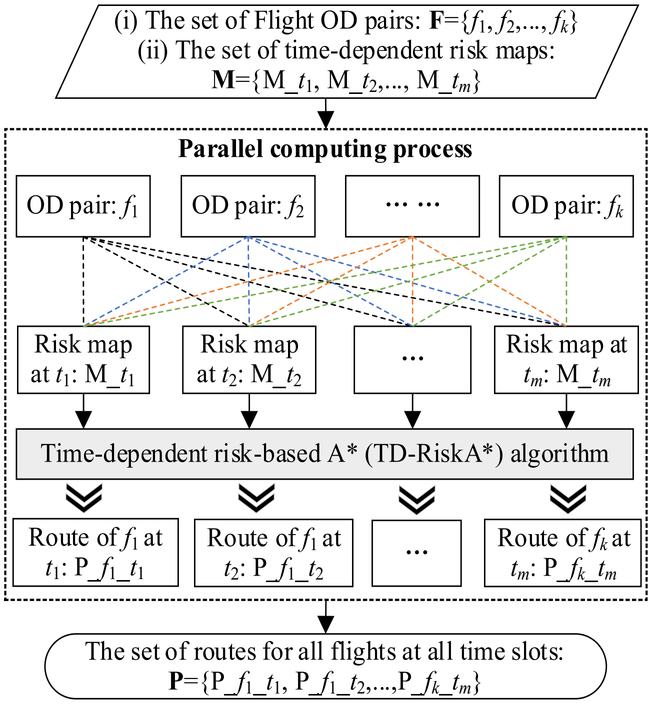


Fig. 5.5: Parallel computing framework to generate the set of candidate routes.

5.4 Two-Stage Stochastic Route Optimization Model

In this section, we develop the Time-Dependent Risk-based Route Optimization (TD-RRO) problem as a two-stage stochastic programming optimization model. In the first stage, the feasibility decision is made to determine if the generated route at a particular time meets the TLOS and flight duration requirements. While in the second stage, route selection and time-related flight decisions are made for the feasible routes to further reduce ground risk cost. The deterministic TD-RRO model is first developed, followed by its stochastic form of TD-RRSO, in which the concept of risk scenario is introduced to incorporate dynamic risk uncertainties.

5.4.1 Deterministic Time-Dependent Risk-based Route Optimization (TD-RRO) model

Under deterministic flight environments, the ground risk costs are determined values. The feasibility and route selection decisions are conducted concurrently to minimize the total ground risk cost for all flight missions. The route set for each OD pair k is pre-generated by using TD-RiskA* algorithm as we developed in Section 5.3.3.

Decision variables

(i). Route feasibility decision x_k. A binary variable to decide the feasibility of the generated flight route.

$$\chi_k = \begin{cases} 1, \text{if there is a feasible route } p_k \text{ for OD pair } k, \\ 0, \text{otherwise} \end{cases}$$

(ii). Route selection $\chi_{p_{k\_}v_i}$. A binary variable to decide if a flight f_k, with departure time t_k falls in the period $\tau_m = [L_m, U_m], m \in \mathrm{M}$, travels on the optimal route among the time-dependent candidate routes. Note that p_k is a feasible and optimal route for OD pair k. The time interval τ_m corresponds with the period of the risk scenario, which defines the risk map M_t_m.

$$\chi_{p_{k\_}v_i} = \begin{cases} 1, \text{if an OD pair } k \text{ travels on route } p_k = \{v_i, i = 1,2, \dots, l\}, \\ 0, \text{otherwise} \end{cases}$$

$$\chi^t_{p_{k\_}v_i} = \begin{cases} 1, \text{if the departure time of } p_k \text{ falls in } \tau_m = [L_m, U_m], \\ 0, \text{otherwise} \end{cases}$$

(iii). Departure time flexibility $t^{\mathrm{flex}}_{f_k}$. An appropriate postponement of departure time can help avoid high-risk periods to reduce the ground risk. The departure time flexibility is denoted as $t^{\mathrm{flex}}_{f_k}$, which is a continuous variable. The initial and actual departure time are then denoted as $t^{\mathrm{dep}}_{f_k}$ and $t'^{\mathrm{dep}}_{f_k}$, respectively. The correlation between them is $t'^{\mathrm{dep}}_{f_k} = t^{\mathrm{dep}}_{f_k} + t^{\mathrm{flex}}_{f_k}$.

Objective function

The overall objective of the TD-RRO model is to minimize the total ground risk cost of all flight OD pairs, which is defined as

min:

$$\begin{aligned} C_{\mathrm{total}} &= \sum_{p_k \in \mathbb{P}} \sum_{v_i \in p_k} c_{v_i} \chi^t_{p_{k_{v_i}}} = \sum_{p_k \in \mathbb{P}} \sum_{v_i \in p_k} \eta_c d_{v_{i_t}} \chi^t_{p_{k_{v_i}}} \\ &= \sum_{p_k \in \mathbb{P}} \sum_{v_i \in p_k} \eta_c \mathcal{F}(t'^{\mathrm{dep}}_{f_k}) \chi^t_{p_{k\_}v_i} \end{aligned} \tag{5.17a}$$

where

$$c_{v_{i\_}t} = \lambda_{\mathrm{LOC}} N^{\mathrm{crash}}_{v_{i\_}t} P_{\mathrm{fatal}} = (\lambda_{\mathrm{LOC}} A_{\mathrm{crash}} S_c P_{\mathrm{fatal}}) d_{v_{i\_}t} = \eta_c d_{v_{i\_}t} \tag{5.17b}$$

$$d_{v_{i\_}t} = \mathcal{F}\left(t'^{\mathrm{dep}}_{f_k}\right), v_i \in p_k \tag{5.17c}$$

Here C_{total} is the total ground risk cost. The constant variables $\lambda_{\mathrm{LOC}}, A_{\mathrm{crash}}, S_c$, and P_{fatal} are converted to η_c. While the ground risk is transformed as population density, which is the function of the actual time ${t'}_{f_k}^{\text{dep}}$ of departing from the location $v_i \in p_k$, where p_k is the selected route for the flight f_k. The function $\mathcal{F}(t)$ is the regression model derived in Section 5.3.2 to estimate the population density.

Constraints

(i). Feasibility requirements. This constraint is to ensure that the pre-generated route for OD pair k is feasible and airworthy in terms of flight duration and TLOS requirements. That is because the operation of electric-powered eVTOLs is highly constrained by the flight duration [187], and by the stringent safety requirement of TLOS [9].

$$\sum_{k \in K} \chi_k = 1, \qquad \chi_k = \{0,1\}, k \in K \tag{5.18}$$

Note that $\chi_k = 1$, if the following conditions hold

$$\sum_{v_i \in p_k} l_{v_i} \leq L_{f_k}, \forall f_k \in F \tag{5.19}$$

$$C_{P_{k\_}t} = \sum_{v_i \in p_k} \eta_c f\left({t'}_{f_k}^{\text{dep}}, \mu_{v_{i\_}t}, \sigma_{v_{i\_}t}\right) \leq \lambda_{\text{TLOS}}, \forall k \in K, t \in T \tag{5.20}$$

where Eq. (5.19) defines the flight duration feasibility, and Eq. (5.20) defines the TLOS feasibility requirement.

(ii). Route selection constraints. Ensure that the flight travels on the selected feasible route,

$$\sum_{p_k \in P} \chi_{p_{k\_}v_i} = \chi_k, \forall v_i \in p, k \in K \tag{5.21}$$

$$\chi_{p_{k\_}v_i} = \{0,1\}, \forall v_i \in p, k \in K \tag{5.22}$$

Ensure the route p_k that flight travelling on is the time-dependent feasible route with departure time t falls in the time interval $\tau_m = [L_m, U_m]$,

$$\sum_{t \in \tau_m} \chi_{p_{k\_}v_i}^t = \chi_{p_{k\_}v_i}, \forall v_i \in p, p_k \in \mathbb{P}, m \in M \tag{5.23}$$

$$\chi_{p_{k\_}v_i}^t = \{0,1\}, \forall v_i \in p, p_k \in \mathbb{P}, t \in T \tag{5.24}$$

(iii). Departure time flexibility constraint. Ensuring the postponed flight time is within the threshold T_{flex}, which presents the maximum tolerable waiting time for a flight plan f_k, denoted as

$$t_{f_k}^{\text{flex}} \leq T_{\text{flex}}, \forall k \in K \tag{5.25}$$

$$t'^{\text{dep}}_{f_k} = t_{f_k}^{\text{dep}} + t_{f_k}^{\text{flex}}, \forall k \in K \tag{5.26}$$

$$\forall\, t_{f_k}^{\text{flex}}, t_{f_k}^{\text{dep}} \in \mathbb{Q}_+, T_{\text{flex}} \in \mathbb{Z}_+ \tag{5.27}$$

5.4.2 Stochastic Model Considering Risk Scenarios: TD-RRSO

To cope with stochastic ground risk conditions due to population movement, a time-dependent risk-based route stochastic optimization (TD-RRSO) model is formulated as a two-stage stochastic optimization program. In the model, the risk-based route planning decisions are made based on the statistical analysis of the expected ground risk conditions, while the feasibility, route selection, and time-related flight decisions are made based on predicted risk conditions, which are characterized by risk scenarios.

Definition 3 (*Risk scenario ω*): Stochastic ground risk conditions for UAS operations are defined as a finite set of risk scenarios Ω, where any risk scenario $\omega \in \Omega$. Under a scenario ω, the stochastic ground risk condition is represented by a vector with random variables as

$$\xi(\omega) = \left(d_{v_{i\_}t}(\omega), \mu_{v_i}(\omega), \sigma_{v_i}(\omega), t_{f_k}^{\text{dep}}(\omega)\right) \tag{5.28}$$

In the two-stage stochastic model, the risk-based routing decision $\mathbf{x} = (x, y, z)$ is made in the first stage, which is to generate the route set at different time points according to historical ground risk conditions. While the route selection and time-related flight decisions $\mathbf{y}(\omega) = \left(\chi_k(\omega), \chi_{p_{k\_}v_i}(\omega), \chi^t_{p_{k\_}v_i}(\omega), t_{f_k}^{\text{flex}}(\omega),\ t'^{\text{dep}}_{f_k}(\omega)\right)$ are made for a given scenario $\omega \in \Omega$ in the second stage to determine: (i) which candidate route is selected to fly at a particular ETD, (ii) when exactly to depart with consideration of time flexibility.

The two-stage stochastic model is formulated as

$$\min \mathbb{E}_\xi[h(\mathbf{x}, \xi)] \tag{5.29}$$

s.t.

$$\sum_{k \in K} \chi_k = 1 \tag{5.30}$$

$$\chi_k = \{0,1\}, k \in K \tag{5.31}$$

where for each risk scenario $\omega \in \Omega$,

$$h(\boldsymbol{x}, \xi(\omega)) = \min_{\boldsymbol{y}(\omega)} \sum_{p_k \in \mathbb{P}} \sum_{v_i \in p_k} \eta_c f\left({t'}_{f_k}^{dep}(\omega)\right) \chi_{p_{k\_}v_i}^t(\omega) \tag{5.32}$$

s.t.

$$\sum_{p_k \in P} \chi_{p_{k\_}v_i}(\omega) = \chi_k, \forall v_i \in p, k \in K \tag{5.33}$$

$$\sum_{t \in \tau_m} \chi_{p_{k\_}v_i}^t(\omega) = \chi_{p_{k\_}v_i}(\omega), \forall v_i \in p, p_k \in \mathbb{P}, m \in M \tag{5.34}$$

$${t'}_{f_k}^{\mathrm{dep}}(\omega) = t_{f_k}^{\mathrm{dep}}(\omega) + t_{f_k}^{\mathrm{flex}}(\omega), \forall k \in K \tag{5.35}$$

$$t_{f_k}^{\mathrm{flex}}(\omega) \le T_{\mathrm{flex}}, \forall k \in K \tag{5.36}$$

$$C_{P_{k_t}} = \sum_{v_i \in p_k} \eta_c f\left({t'}_{f_k}^{\mathrm{dep}}(\omega)\right) \le \lambda_{\mathrm{TLOS}}, \forall k \in K, t \in T \tag{5.37}$$

$$\chi_{p_{k\_}v_i}(\omega) = \{0,1\}, \forall v_i \in p, k \in K \tag{5.38}$$

$$\chi_{p_{k\_}v_i}^t(\omega) = \{0,1\}, \forall v_i \in p, p_k \in \mathbb{P}, t \in T \tag{5.39}$$

$$\forall\, t_{f_k}^{\mathrm{flex}}(\omega), t_{f_k}^{\mathrm{dep}}(\omega) \in \mathbb{Q}_+, T_{\mathrm{flex}} \in \mathbb{Z}_+ \tag{5.40}$$

The combined form of the two-stage stochastic optimization model can be represented as

$$\min\ \mathbb{E}_\xi[h(\mathbf{x}, \xi)] = \sum_{\omega \in \xi} \mathcal{P}_\omega \left(\sum_{p_k \in \mathbb{P}} \sum_{v_i \in p_k} \eta_c f\left({t'}_{f_k}^{\mathrm{dep}}(\omega)\right) \chi_{p_{k\_}v_i}^t(\omega) \right) \tag{5.41}$$

s.t.

Constraints (30)-(40).

where $\mathcal{P}_\omega$ is the probability of each risk scenario, and $\sum_{\omega=1}^{\xi} \mathcal{P}_\omega = 1, \mathcal{P}_\omega \in [0,1]$.

Note that the decision variables include both binary ones ($\chi_k, \chi_{p_{k\_}v_i}$, and $\chi_{p_{k\_}v_i}^t$) and continuous variables ($t_{f_k}^{\mathrm{flex}}$ and ${t'}_{f_k}^{\mathrm{dep}}$). The objective function (Eq. (5.32)) and Constraint (Eq. (5.37)) are nonconvex and nonlinear functions derived from the regression model as depicted in Fig. 5.4. That makes the optimization problem a nonconvex mixed integer nonlinear programming (MINLP) model, which is NP-hard

[188]. The recourse matrix is random in both stages, making the problem even more difficult to solve.

To handle stochastic programming (SP) problems, several groups of approaches have been proposed. One group is decomposition methods. Benders decomposition has made the first significant progress in solving SP problems with continuous recourse [189]. Follow-up works extended the decomposition approach to solve stochastic integer programming problems based on the L-shaped method [190], branch and bound [191], split cuts [192], and dynamic programming [193]. However, the decomposition approaches require considerable computational effort to obtain lower bounds in dual decomposition, which is inefficient in solving large-scale problems with fast convergence speed [177].

These computational challenges motivate the use of meta-heuristic evolutionary algorithms, as they are not sensitive to the problem size and have little cost to optimality loss [141], [177], [194]. The analysis of convergence guarantee for meta-heuristic algorithms forms a theoretical stand [195] and makes them a primary alternative for solving large-scale global optimization problems. As the main interest of this work is not on the solving methods of the MINLP problem, an effective and well-proven meta-heuristic algorithm proposed in our previous work [141], [194] is used to solve the developed nonconvex MINLP problem in this Chapter.

5.5 Numerical Analysis and Discussion

We implement our proposed optimization model using real-world public transportation data and census population density data to demonstrate the model's performance in terms of ground risk assessment and mitigation. Computations were conducted using Matlab software at version 2020a on a workstation with an eight-Core Intel i9 CPU (2.5 GHz) and 32 GB RAM.

The simulation environment is selected from an area in Singapore with a size of 6,000m×6,000m×120m (width × length × altitude), which is meshed into standard 3D blocks with a size of 100m×100m×30m. The census population density of the selected environment is 8.358×10^3 (people/km^2), and the basic parameters (defined as η_c in Eq. (5.17b)) used in the risk assessment model are referred from a previous research work [170]. The public transportation data is taken from one subway station in Singapore with historical passengers' tape-in and tape-out data. The data is taken as hourly weekday average number of passengers, and 52 weeks of data are collected to develop the population density estimation model.

5.5.1 Time-Dependent Candidate Routes under Different Risk Scenarios

5.5.1.1 Generation of Risk Scenario

In this subsection, we generate the time-dependent risk-based routes for a given OD pair under different risk scenarios. The risk scenarios are defined and generated by incorporating historical population movement data (as shown in Section 5.3.2) and the risk levels used in UAS operations [27], [196], as shown in Table 5.1. Within one scenario, the distribution of uncertain parameters (i.e., population movement conditions in this work) remains at the same level. For instance, in the negligible risk scenario (S1), the population movement is almost inconceivable, which corresponds to midnight hours. The fatality risk makes no difference at 01:00 am or 04:00 am in terms of risk-based route optimization. While it makes a difference across different scenarios due to significant changes in population density.

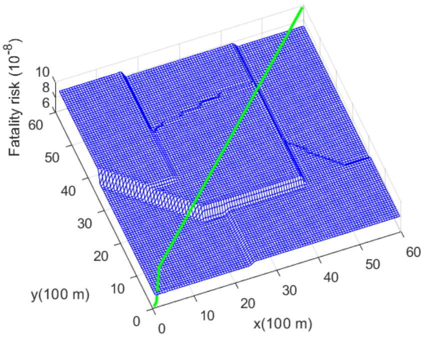
Table 5.1: Scenario definition based on UAS operational risk levels associated with population density.

Scenario (Risk level)	Description
S1 (Negligible)	Fatality risk is negligible, as the population movement is inconceivable (00:00-05:00)
S2 (Low)	Fatality risk is low, as population density is sparse (05:00-07:00, 21:00-24:00)
S3 (Medium)	Fatality risk is medium, as frequent population movement (09:00-18:00)
S4 (High)	Fatality risk is high, as population density is at its peak (07:00-09:00, 18:00-20:00)

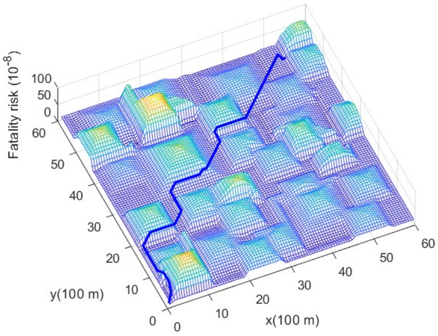
5.5.1.2 Candidate Routes under Different Scenarios

According to the generated scenarios, population density distributions are computed and used for the generation of corresponding risk maps. The maps are then used to produce time-dependent risk-based routes using TD-RiskA* algorithm for a given OD pair under each scenario. The produced risk-minimizing candidate routes under each scenario are presented in Fig. 5.6.

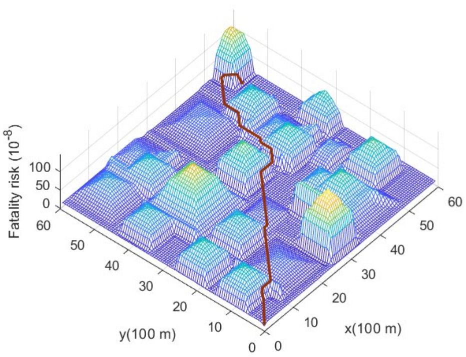
Fig. 5.6(a)-(d) demonstrates that the fatality risk distribution derived from population density is significantly different across different risk scenarios. That subsequently results in different optimal routes (measured by risk cost) for the same OD pair, as shown in Fig. 5.6(e). Under the negligible risk scenario, the population density exposed to the risk of UAS operations is approximately zero. The obtained risk-minimizing route under this scenario is the same as the distance-minimizing route, which directly flies from origin to destination. For other scenarios, the fatality risk increases with the increase of the risk level. The obtained routes avoid high-risk areas to reduce the total ground risk cost, while the distance of these routes shows an increased trend compared with Scenario 1, as these routes need to travel longer distances to avoid high-risk areas.



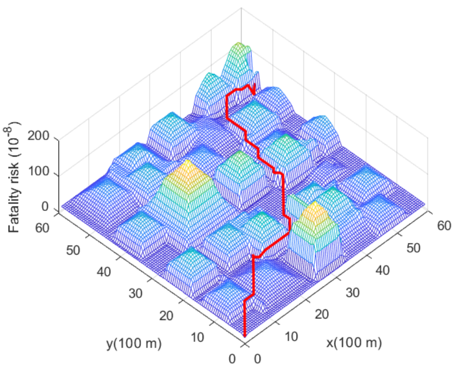
(a) S1: Negligible risk scenario



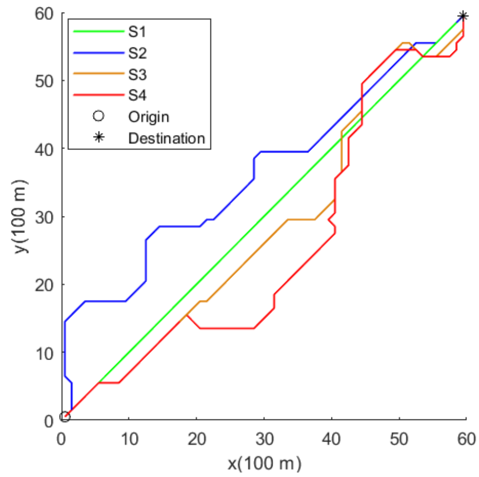
(b) S2: Low risk scenario



(c) S3: Medium risk scenario



(d) S4: High risk scenario



(e) Top view of risk-based routes under different scenarios

Fig. 5.6: Time-dependent risk-based candidate routes for a single OD pair under different risk scenarios.

The above results demonstrate the spatial-temporal differences of the risk conditions for UAS operations, which suggests that the route optimization should consider both the actual departure time (operating in which scenario) and the risk condition (how exactly to fly in a scenario). Joint decision-making is conducted using our proposed TD-RRO model for both two factors, and the performances are presented in the following section.

5.5.2 Optimization Performance of TD-RRO model

5.5.2.1 Time-Dependent Risk Optimal Routes

A toy problem with a small scale of OD pairs is first solved to demonstrate the decision-making process of the TD-RRO model. We then solve a large-scale problem in comparison with the distance-based (DB) model [197] and static risk-based (RB) model [85] in terms of risk cost and flight distance. The OD pairs are randomly generated with coordinates of origin and destination selected from the range of $x, y \in [1,60], z \in [1,4]$. The estimated time of departure (ETD) is randomly assigned to the OD pairs in the range of [1, 24*60=1440] minutes.

In the toy problem, six OD pairs are randomly generated. The set of candidate routes for each OD pair is produced using TD-RiskA* algorithm, shown in color lines in Fig. 5.7. With the set of candidate routes, the TD-RRO model is employed to obtain the time-dependent optimal route for each OD pair by conducting the feasibility decision (defined by Eqs. (5.18)-(5.20)) and route selection decision (Eqs. (5.21)-(5.24)). Obtained results are presented as the dotted black line.

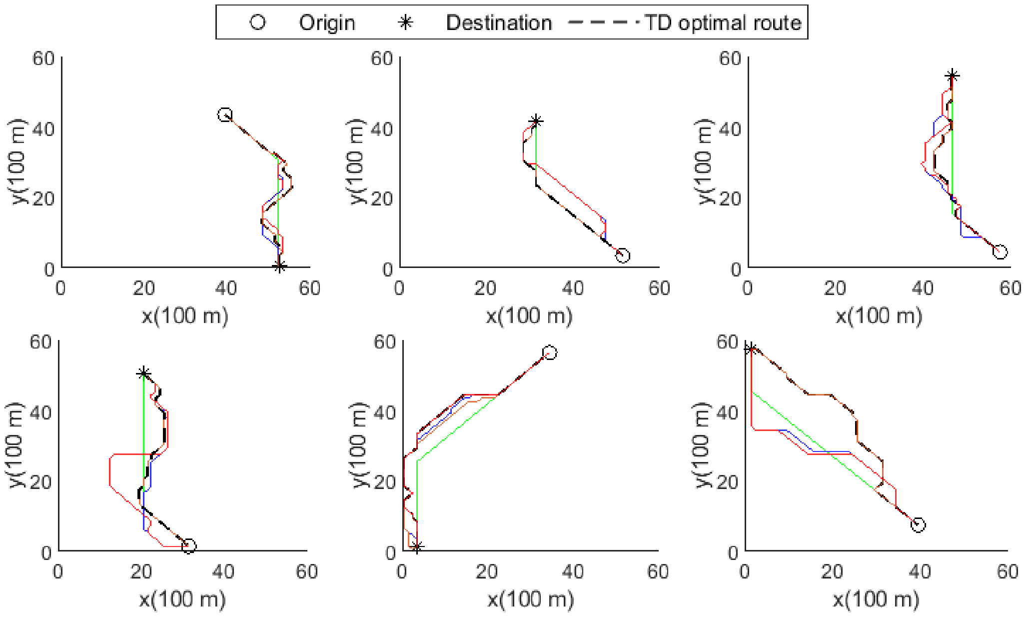


Fig. 5.7: Set of candidate routes with the selected time-dependent optimal route for different OD pairs.

To demonstrate the generalizability of the model, a large-scale case is created with 100 randomly generated OD pairs. Comparisons are made among three different route planning methods: (i) Distance-based (DB) A* method, (ii) Conventional risk-based (RB) method RiskA*, and (iii) our method TD-RRO model. The ground risk of obtained optimal routes by the three methods is evaluated and given in Fig. 5.8 and Table 5.2.

Our proposed TD-RRO method performs best in reducing ground fatality risk (by 36.13% on average), while the average risk cost of a flight mission in all scenarios meets the TLOS requirement ($<10^{-6}$), as shown in Table 5.2. Followed by the risk-based (RB) method, which significantly reduces the risk cost (by 28.82% on average) compared with the distance-based method, as it avoids the empirical high-risk areas. However, in the high-risk scenario (S4), the average risk cost (1.25×10^{-6}) exceeds the TLOS requirement, making flight missions at S4 infeasible. That is because the empirical risk map used in the RB method cannot fit all scenarios where the risk conditions are different in spatial-temporal dimensions. Our proposed TD-RRO method, on the other hand, makes joint decisions to determine the optimal route for flight missions according to the risk scenarios they belong to, which selects the most suitable route for each flight mission.

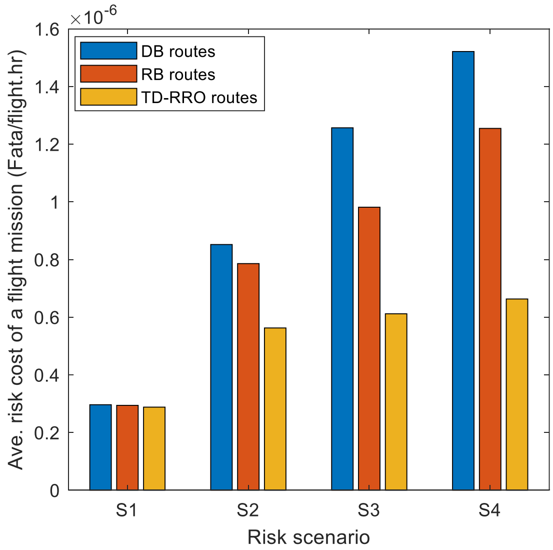


Fig. 5.8: Our proposed TD-RRO method performs best in ground risk reduction at all risk scenarios compared with distance-based (DB) and risk-based (RB) methods (by 36.13% and 28.82% on average, respectively).

Viewing the performances from individual scenarios in Table 5.2, significant risk mitigations are made in medium-risk scenario S3 (51.33%) and high-risk scenario S4 (56.41%), which are at periods of frequent population movement. Fatality risks at some locations with train/bus stations are remarkably high. Avoiding these locations can significantly reduce the risk cost of a flight route, while failing to do so can result in unbearably high-risk cost as demonstrated by the routes generated by the DB method in S3 and S4 scenarios. In negligible risk scenario S1, the average risk cost of the routes has no significant difference (2.81%), as the risk map is generated by using census population density, which is a constant for the whole area. Note that the average risk cost of the routes by all three methods increases with the change of scenario from S1 to S4, as the risk level increases. This means operating in periods associated with low-risk scenarios can reduce the risk cost, suggesting that decision-makers can better arrange their flight plans at different periods of the day to increase the flight approval (feasibility) rate and reduce the overall ground risk cost.

The optimal routes generated by the TD-RRO method have an efficiency loss in flight distance, as shown in Fig. 5.9 and Table 5.3. The routes produced by risk-based approaches (TD-RRO and RB) have longer flight distance (by 7.87% and 8.56%, respectively), compared with the DB method. That is because the flight will have to travel a longer distance to avoid high-risk areas. While for the routes generated by TD-RRO and RB methods, there is no significant difference (0.15%) in terms of flight distance. Based on obtained results in this case, we demonstrate that the TDRB routes generated by TD-RRO significantly reduce the fatality risk by 36.13%, while it has little cost to operational efficiency by 7.87% in terms of flight distance increase.

Table 5.2: Average ground risk (Fatality/flight.hr) and comparison of our TD-RRO method with DB and RB methods.

Scenarios	DB	RB	TD-RRO	TD-RRO vs. DB	TD-RRO vs. RB
S1 (Negligible)	2.96×10^{-7}	2.94×10^{-7}	2.88×10^{-7}	-2.81%	-2.06%
S2 (Low)	8.52×10^{-7}	7.86×10^{-7}	5.63×10^{-7}	-33.96%	-28.40%
S3 (Medium)	1.26×10^{-6}	9.81×10^{-7}	6.12×10^{-7}	-51.33%	-37.67%
S4 (High)	1.52×10^{-6}	1.25×10^{-6}	6.63×10^{-7}	-56.41%	-47.15%
Average	9.82×10^{-7}	8.29×10^{-7}	5.31×10^{-7}	-36.13%	-28.82%

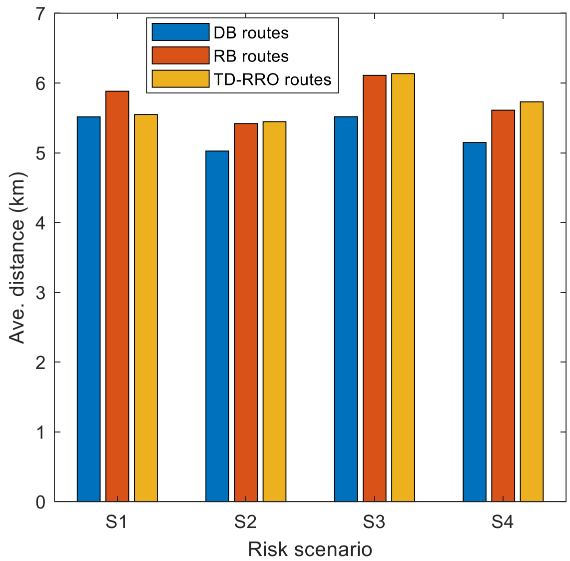


Fig. 5.9: The efficiency loss (measured by distance) of routes generated by the TD-RRO method is acceptable, with an average of 7.87% longer than that of the distance-based method.

5.5.2.2 Optimal Routes with Departure Time Flexibility (DTF)

A. Optimal Routes vs. Optimal Routes with DTF

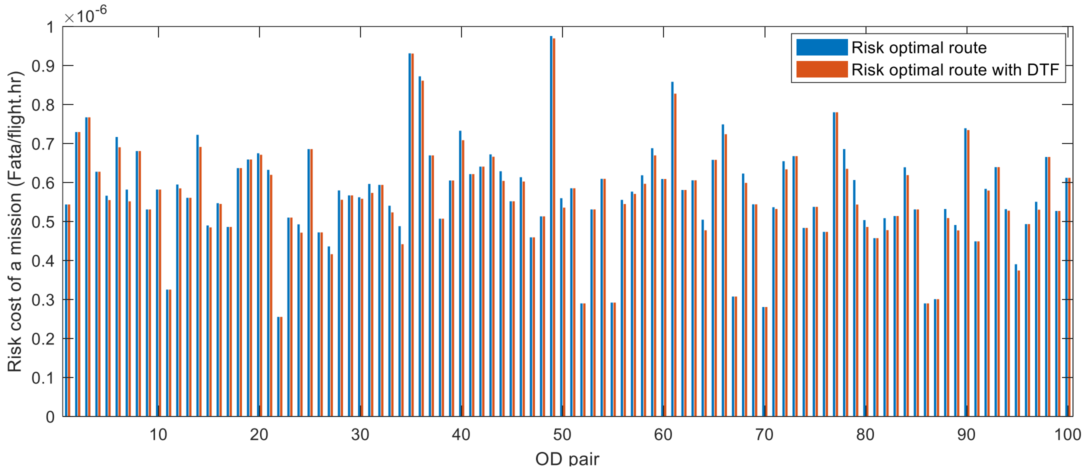
Above, we demonstrate the performance of the time-dependent risk-based routes generated by the TD-RRO model, which conducts the route feasibility and selection decisions. This section analyses the performance of optimal routes with departure time flexibility (DTF), which is another decision variable to further reduce risk cost by adjusting the actual departure time, defined in Eq. (5.17c). The DTF is considered the waiting time of a flight plan to avoid high-risk periods so as to reduce operational risk costs. The simulation environment and flight plan with 100 OD pairs use the same settings described in Section 5.5.1. The estimated time of departure (ETD) is randomly assigned to each OD pair in the range of [1, 1440] minutes, and the threshold of DTF is taken as 15 minutes. The selected risk optimal route for each OD pair is used as the basis for the comparison with the optimal route with DTF. Obtained results are presented in Fig. 5.10.

Table 5.3: Average ground risk (Fatality/flight.hr) and comparison of our TD-RRO method with DB and RB methods.

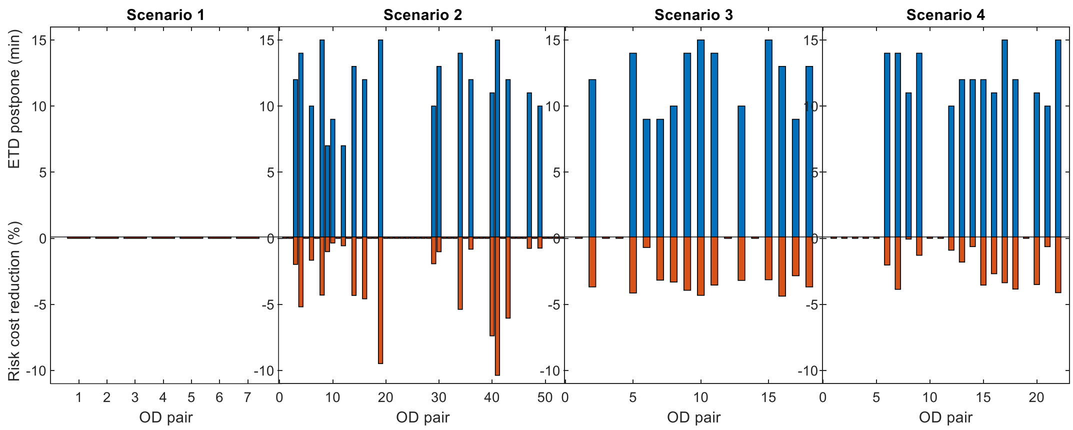
Scenarios	DB	RB	TD-RRO	TD-RRO vs. DB	TD-RRO vs. RB
S1 (Negligible)	5515.40	5881.93	5547.70	0.59%	-5.68%
S2 (Low)	5024.99	5418.09	5445.58	8.37%	0.51%
S3 (Medium)	5516.91	6109.62	6134.15	11.19%	0.40%
S4 (High)	5146.81	5610.32	5730.49	11.34%	2.14%
Average	5301.03	5754.99	5714.48	7.87%	-0.66%

The optimal route with time flexibility performs better than that of the original optimal routes in terms of ground risk reduction, as shown in Fig. 5.10(a). Notable risk reductions are seen in OD pairs (e.g., OD-14, OD-61, and OD-78). That indicates a postponement of departure time (DTF) contributes to the risk cost reduction, as the risk conditions in a given scenario may become better at new ETD (for instance, a postponed departure in S4 could avoid high-risk conditions at peak hours). While a number of OD pairs (54 in total) show no risk reduction because the risk condition may become even worse after the postponement of these flights.

To have a deeper understanding of this situation, we further study the correlations among departure time flexibility, risk scenario, and risk reduction. Obtained results are given in Fig. 5.10(b). Seven routes are generated in S1, and none of them has had the risk cost reduced (0%, as shown in Table 5.4). As in S1, the risk condition remains constant due to inconceivable population movement. The ETD of a flight will not be postponed for risk reduction if the risk cost has no space to reduce, which makes the optimized postponement of ETD for each OD pair zero. In Scenario 2 (low risk scenario), the risk cost of a large portion of OD routes (35.85%) is reduced by DTF within the 15-minute threshold. That portion further increases to 72.22% and 63.64% in S3 and S4, respectively. Because in and around peak hours of population movement in S3 and S4, a slight ETD postponement helps avoid high-risk periods to reduce ground risk cost. While the percentage of risk reduction decreases with the increase of scenario risk level from S2 (18.36%), S3 (16.82%) to S4 (11.19%). This is because the fundamental risk cost of a given route increases with the rise of risk level from S2 to S4, as demonstrated in Fig. 5.8.



(a) Departure time flexibility (DTF) has contributions to ground risk reduction



(b) Risk reduction contributed by DTF at each scenario

Fig. 5.10: Performance of departure time flexibility (DTF) in ground risk reduction.

Obtained results demonstrate that the departure time flexibility can reduce risk cost by 11.59% on average, with a success rate of 42.93% under all scenarios. It also suggests that the ETD postponement decision should be made case-by-case, as not all flight plans are suitable for postponing their initial ETD. That further emphasizes the significance of our proposed TD-RRO model, which enables automatic decision-making of DTF for safe UAS operations.

Table 5.4: Average risk reduction per departure time flexibility (DTF) hour.

Scenarios	S1 (Negligible)	S2 (Low)	S3 (Medium)	S4 (High)	Average
Risk reduction	0%	18.36%	16.82%	11.19%	11.59%
Rate of postpone	0.00%	35.85%	72.22%	63.64%	42.93%

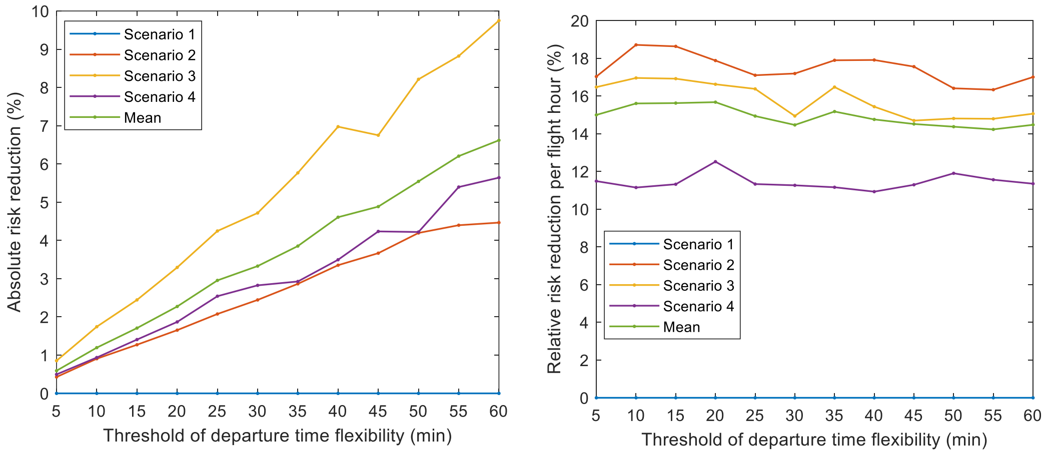
Note that the unit of average risk reduction is converted from risk cost per flight mission to risk cost per flight hour for practical uses, as the unit of TLOS is fatality per flight hour. For instance, the risk reduction for a 15-minute (1/4 hour) flight route is 5%, and the converted risk reduction is 20% per flight hour for this route.

B. Sensitivity of DTF threshold

We study the sensitivity of the departure time flexibility (DTF) threshold, which determines the upper bound of the ETD postponement. The threshold of DTF is taken in the range of [5, 60] minutes with an interval of 5 minutes. Sensitivity analysis results are shown in Fig. 5.11.

Absolute risk cost reduces linearly with the increase of the DTF threshold in all scenarios except for S1, as shown in Fig. 5.11(a). That is because a larger threshold allows more ETD postpones avoiding high-risk periods. For instance, the initial ETD of a flight is 7:30 pm, and the postponement of ETD can avoid high-risk hours if the threshold is more than 30 minutes (new ETD becomes 8:00 pm onwards). For individual scenarios, there is no risk reduction in S1 by postponing ETD, while the absolute risk reduction in S3 is the most notable one. As S3 is the starting or closing of high-risk periods, ETD postponement can effectively help flights avoid operating in high-risk periods. S2 and S4 correspond to risk conditions that last for a long period of time, and an ETD postponement may not be able to convert them to another low-risk condition.

To further analyze the effectiveness of ETD postponement, we study relative risk reduction, which is defined as the risk reduction of a flight route per flight hour. As shown in Fig. 5.11(b), the relative risk reduction shows no significant changes with the increase of the DTF threshold, which corresponds to the linear trend in the absolute risk reduction, as shown in Fig. 5.11(a). Meaning that the unit gain of ETD postponement remains the same. Decision-makers will have to cast the same magnitude of delays to obtain the risk reduction for a flight plan. Besides, the relative risk reduction under individual scenarios shows a different pattern. The S2 has the most relative risk reduction (18%) among all scenarios, whereas its absolute risk reduction is the second least. That is because the fundamental risk cost for a given flight route increases with the risk of risk condition from S2 to S4. The fundamental risk cost acts as a denominator, which decreases the relative risk cost from S2 to S4 as demonstrated in Fig. 5.11(b).



(a) Absolute risk reduction over total time (b) Relative risk reduction per unit time

Fig. 5.11: Ground risk cost reduction with the increase of departure time flexibility (DTF) threshold. Note that absolute risk reduction is defined as the risk reduced with a total DTF. While the relative risk reduction is defined as the risk reduced per DTF hour.

5.5.3 Benefits of Stochastic Optimization by TD-RRSO Model

In this subsection, we assess the benefits of stochastic optimization results obtained by our TD-RRSO model by introducing the value of the stochastic solution (VSS) and the expected value of perfect information (EVPI). The impact of route selection flexibility under stochastic risk conditions is investigated. The risk conditions associated with population density are categorized into four scenarios based on the risk level classification rule in the civil aviation field [27], [196], which is given in Table 5.1. The risk condition, specifically the risk cost of each grid cell in each scenario, is represented as risk maps, depicted in Fig. 5.7(a)-(d). We assume the probability of each scenario (defined in Eq. (5.41)) is equal, which is taken as 25%.

To assess the VSS and EPVI, three types of problems needed to be solved under the defined scenarios, and they are: (i) the solution of the recourse problem (RP), also known as here and now solution, which is the optimal solution obtained by the TD-RRSO model in this work, (ii) wait and see (WS) solution from a deterministic problem based on perfect information, which corresponds to the solution obtained by our proposed deterministic TD-RRO method, and (iii) the expected value (EEV) solution resulting from a deterministic problem with replacing all random variables by a single nominal scenario. With these definitions, the VSS and EPVI are obtained as VSS=EEV-RP and EVPI= RP-WS [171]. After computation, the obtained results are presented in Fig. 5.12 and Table 5.5.

The WS solution, not surprisingly, performs best with the lowest risk cost for all flight routes, as it has perfect information. While the RP solution considering stochastic risk conditions achieves the second-best performance with all generated flight routes that met the TLOS requirement (<10^{-6} fatalities/flight.hr). However, the EEV performs the worst, with 18 flight routes exceeding the TLOS requirement under stochastic risk conditions. That is because EEV only considers a single expected risk condition to make decisions for route selection in all risk scenarios.

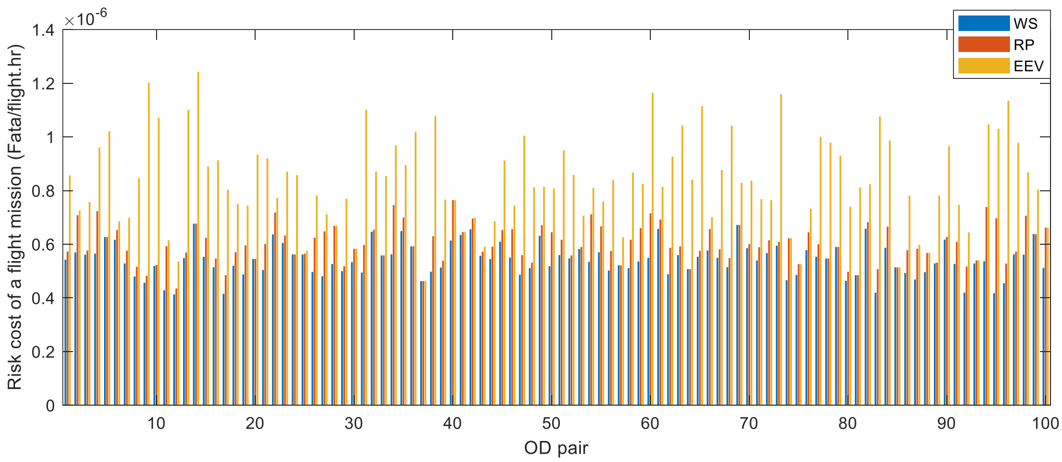


Fig. 5.12: Comparison of wait and see (WS), recourse problem (RP), and expected value (EEV) solutions.

With the WS, RP, and EEV results, we obtained the VSS and EVPI, as shown in Table 5.5. The VSS quantifies the benefits of our stochastic optimization model TD-RRSO compared with a deterministic solution that directly optimizes first-stage and second-stage solutions with a single scenario (i.e., considering the average risk condition among all scenarios). The EVPI evaluates the residual cost of the risk uncertainty originating from the fact that the first-stage feasibility decisions need to be made before the uncertainty is solved, which marks the key difference with the independent decision-making in the first-stage and second-stage, separately. Based on the results obtained in our case, the VSS is the reduction of 38.96% of risk cost, while the risk cost can be further reduced by 11.77% if perfect information can be obtained.

Table 5.5: Value of stochastic optimization and expected value of perfect information.

Indicators	Average risk cost at all scenarios					Risk reduction	
	WS	RP	EEV	VSS	EVPI	VSS	EVPI
Results	6.07E-08	6.75E-08	9.26E-08	2.52E-08	6.72E-09	38.96%	11.77%

5.6 Summary

This Chapter studies the stochastic route optimization problem under dynamic ground risk for safe UAS operations in urban environments. The problem is developed as a two-stage stochastic route optimization model. The first-stage decision handles the route feasibility requirements regarding the target level of safety and flight duration. While the route selection and departure time-related decisions are made in the second stage to cope with stochastic risk conditions. Simulation results demonstrate the good performance of the proposed model in terms of ground risk mitigation. The main conclusions of this work are summarized as follows.

(i) The developed time-dependent risk assessment model allows quantitative evaluation of the dynamic ground risk for specific UAS operations. That enables quantitative feasibility evaluation for specific flight plans with respect to the target level of safety at different departure times. This finding is used in the first stage of the stochastic optimization model to decide the feasibility of the flight routes.

(ii) The proposed time-dependent risk-based route planning method TD-RiskA* with parallel computing scheme enables large-scale route set generations in polynomial time, which prepares the route selection decision in the second stage of the stochastic route optimization model.

(iii) Under a deterministic risk condition, the developed time-dependent risk-based route optimization (TD-RRO) model can significantly reduce the ground fatality risk (by 36.13%) while having only little cost to operational efficiency (by

7.87%) in terms of flight distance increase, compared with the routes generated by the distance-based method.

(iv) Under uncertain risk conditions, our proposed TD-RRSO model performs well in risk mitigation, as it jointly optimizes the route feasibility decision (before the uncertainty is resolved) and route selection and departure time decisions (after the uncertainty is resolved). The benefits of the stochastic model are quantified by the value of the stochastic solution (VSS) and the expected value of perfect information (EVPI). Based on simulation results, the VSS is the reduction of 38.96% of risk cost, while the cost can be further reduced by 11.77% if perfect information can be obtained.

(v) Departure time flexibility (DTF) contributes to ground risk reduction (by 11.59% on average). However, the DTF decision should be made on a case-by-case basis, as not all flights are suitable for postponing their initial estimated time of departure. That further emphasizes the significance of our proposed TD-RRSO model, which enables automatic decision-making of DTF for safe UAS operations.

The research findings of this work can provide support for (i) aviation authorities to decide if a specific flight plan (with flight route and estimated time of departure) can be approved under a given target level of safety, (ii) eVTOL users (delivery or air taxi companies) to optimize the routes and timetables of their flight missions to avoid high-risk conditions. Which ultimately contributes to the safety risk management of UAS operations in low-altitude urban airspace.

Chapter 6

6 Multi-UAV 4D Trajectory Optimization and Conflict Resolution for Air Risk Mitigation

In Chapter 5, the ground risk assessment and mitigation problems are studied to produce robust and risk-minimizing flight routes for individual operations. However, there are demands (e.g., drone delivery) for multiple flight operations in the same spatial-temporal space, resulting in a high-density UAS traffic flow. The high-density traffic increases the number of potential flight conflicts and delays that need to be addressed.

In this Chapter, flight conflict detection and resolution for air risk mitigation is conducted at the strategic phase to produce conflict-free four-dimensional trajectories (4DT). An adaptive conflict resolution framework is proposed to optimize the strategies used for different types of conflicts with explainable mechanisms and to optimize UAV 4D routes. The framework is formulated as a double-layer optimization problem, and scheduling, speed adjustment and rerouting strategies are considered for conflict resolution. The first layer of the framework is established as a probabilistic selection model to make decisions on which strategy is selected for what type of conflict. The second layer is developed as a mixed integer nonlinear programming (MINLP) model to optimize the decision variables of the strategies selected by the first layer. A novel meta-heuristic stochastic fractal search (SFS) algorithm is then developed to solve the proposed problems. Simulation results demonstrate the effectiveness and robustness of the proposed framework and algorithms.

6.1 Introduction

Air risk of flight conflict is a major challenge for safe UAV operations. The various UAV applications always require operating at the same period in the same airspace, as flight demands (e.g., food delivery and passenger transport) have peaks [198], [199]. That increases the potential flight conflicts. To solve these challenges, the risk-based UAV 4D route optimization with adaptive conflict resolution problem is investigated. By generating risk-based routes, UAVs can avoid not only explicit obstacles but the implicit high-risk areas where the probability of causing fatalities and damages is high. With the obtained risk-based routes, an effective conflict resolution method is required to optimize the pre-planned 4D routes by solving the conflicts using different types of strategies.

The strategies consist of two aspects. The first aspect is from the temporal dimension including the scheduling and speed adjustment strategies. The scheduling is to change the estimated time of departure (ETD) of each flight while the speed adjustment will change the estimated time of arrival (ETA) of each waypoint. The second aspect is from the spatial dimension and the strategy is rerouting, which is to adjust the flight route by changing the coordinates (x, y, z). The commonly used uncertainty models in conflict resolution studies are also discussed in this section.

A. Strategies in temporal dimension: scheduling and speed adjustment

Scheduling is an effective way to tackle pre-flight conflicts. By adjusting the departure time, potential conflicts or congestion can be solved. In civil aviation fields, scheduling problems have been studied to optimize the air traffic flow management system. A 4D strategic deconfliction problem at a continental scale was developed as a mixed-integer programming model to optimize the trajectory interactions [39]. In this study, the departure time of flights can be shifted to solve the potential trajectory interactions. In a follow-up study, the wind-optimal flight trajectories were considered in the strategic deconfliction problem for the North Atlantic oceanic airspace [200]. The objective of this study is to minimize the conflicts by changing the departure time and trajectory, while constraints are given to minimize the deviations of the flight plan from the initial one. Authors also developed mathematical models to maximize the runway throughput [104] as well as to utilize en-route capabilities [106]. A follow-up work [201] considered the demand and supply balance and proposed an integrated flight scheduling framework to efficiently optimize the traffic flow.

Compared with commercial manned aircraft, the scheduling of UAS operations has more challenges, as the UAS traffic network is more complex in low-altitude urban airspace. In recent years, researchers have studied the UAS scheduling problems with various optimization objectives, ranging from minimizing flight conflicts [94], [116], [142] and flight delays [118], [186], to maximize the coverage rate of surveillance missions [111], [112]. To achieve these objectives, various models have been proposed. Mathematical programming is the most used one. Based on different modelling requirements of decision variables, authors [118] developed the mixed-integer linear programming model to optimize the pre-planned routes in discrete environments where the departure time was taken as an integer. In other studies of [185], [199], [202], the models involve energy constraints or payload constraints, which are nonlinear. Optimization problems were developed as mixed-integer nonlinear programming to generate feasible flight routes. To solve the proposed optimization models, both exact methods such as the branch-and-cut algorithm [185], and dynamic programming algorithm [203], and heuristic algorithms such as the auction algorithm [109], and genetic algorithm [118] were developed. In some cases, the feasible region of scheduling problems is nonconvex due to the complex objective or constraint functions [103]. To solve the nonconvex optimization problems for drone scheduling, the authors investigated an approach to transform a nonconvex problem into two convex subproblems without sacrificing the optimality of the solution [111]. They also proposed an exact effective algorithm to solve the transformed convex problems. What is more, some authors paid attention to developing more robust methods and algorithms for scheduling and task assignment [112]. They introduced iterative strategies to improve the performance of the task assignment algorithm and their strategies improved the stability of proposed methods while reducing the computational complexity.

Another conflict resolution strategy is to adjust the flight speed, which subsequently adjusts the ETA for the succeeding waypoints of a flight. The adjustment of ETA changes the time difference between two consecutive flights passing a waypoint. For instance, accelerating the preceding flight while decelerating the succeeding one widens the time difference, which solves a conflict. Speed adjustment is also an effective and commonly used strategy for flight conflict resolution in air traffic management fields. To solve the aircraft conflict resolution problem, a geometric optimization method was developed [204]. The velocity vector strategy was used to change the heading and speed of the aircraft, and the optimized 4D trajectories, therefore, have minimum deviations from the initial ones. In a follow-up study, a subliminal speed control method was proposed to solve the CD&R problems with only minor adjustments of speed [205]. That generated the minimum deviations for the initially-planned trajectories and therefore reduced the air traffic controllers' workload on handling the conflicts [52]. Some research works also studied the robust conflict resolution problems with uncertainties that wind components and velocities of aircraft were randomly generated in a reasonable range [40], [41]. To overcome the uncertainty impact on the pre-planned 4D routes, one study [42] proposed a dynamic adjustment method to adjust the airspeed in response to the leads or lags of flight, which increased adherence to the estimated time of arrival at each waypoint in the pre-planned route. Another work [206] studied the speed approach for UAV conflict resolution, while the critical speed range for potential conflicts was considered to reduce the computational cost. To solve the potential conflicts in UAV formation flight, researchers [121], [207] investigated the speed change strategy based on potential field vectors. The idea of these studies is to generate a conflict-free envelope by adjusting the flight speed of the UAV. A recent study [208] has the same mind to generate conflict-free UAV 4D routes. In that study, the airspace was divided into standardized air blocks where the airspace occupation time and rate were considered. The occupied air blocks were disabled for other routes, which prevents flight conflicts.

B. Strategy in spatial dimension: rerouting

Rerouting is another effective way to tackle flight conflict. By adjusting the 3D geometric trajectory, potential conflicts can be resolved in the spatial dimension. Compared with the scheduling strategy, the rerouting strategy changes the flight trajectory to avoid conflicts, convective weather, and other risky areas [43], [44]. For manned aircraft, rerouting is not a preferred conflict resolution strategy considering safety and efficiency issues. Commercial manned aircraft are only allowed to fly within the airway boundary, and lateral rerouting may deviate the trajectory out of the boundary resulting in risk issues. The altitude change is also a fuel-consuming and unsafe way, which is demonstrated by a number of research works that aim to minimize the trajectory deviations from the initial one when using the rerouting strategy [39], [209]–[211].

From an efficient point of view, minimizing the flight distance is always one of the objectives for solving both manned and unmanned conflict resolution problems using the rerouting strategy. However, it would be more challenging for UAS operations, as rerouting may generate extra flight distance and that may make the mission infeasible for battery-powered UAVs due to its limited flight duration. To solve this problem, some authors [119], [185] proposed the energy-constraint rerouting method to ensure the UAV is able to safely return home. Come to the application of rerouting strategy, it was conducted by using both discrete and continuous methods. In discrete environments, the airspace is meshed into units and the UAV operates from one waypoint to another. Rerouting in that environment is to search for an alternative route point-by-point using methods such as A* [208] and the ant colony algorithm [167]. The advantage of discrete rerouting is that the traffic flow can be better organized via the waypoint-based network, which is good for high-density traffic management [55], [212]. On the other hand, researchers also investigated the rerouting problems using continuous approaches such as vector fields [79], [122]. Their proposed vector field methods considered UAV kinematics and approximate speed modulation, which enabled smooth and conflict-free routes for multi-UAV operations. B-spline curves have also been used to connect each route waypoint and produced continuous conflict-free routes for several application uses [123], [213]. Compared with the discrete method, the continuous methods perform better in terms of local route optimization and route smoothing. Instead of changing the coordinates of the route in three dimensions, some authors investigated the flight-level adjustment method for deconfliction [110]. They proposed a flight-level assignment strategy to solve conflicts for pre-planned 4D routes, and the operational information was considered to determine the assigned flight level.

C. Uncertainties in flight conflict resolution

The common uncertainties considered in conflict resolution models for manned aircraft are weather uncertainties [42], [200], [210], [211], and trajectory prediction uncertainties [39], [40], [205]. The weather uncertainty models are always developed with a probability density function that is derived from the data of weather forecasts for the prevailing wind in large-scale space. While the aircraft trajectory prediction uncertainties are always presented as random variables of velocities and positions in reasonable ranges. However, in the small-scale low-altitude urban airspace, weather conditions are difficult to predict (e.g., wind shear in the landing phase for manned aircraft), which makes it difficult to develop the weather uncertainty model for UAS operations. The trajectory uncertainties for UAS operations are also preferred to be solved in the tactical phase at robotic levels [214]–[216].

As discussed above, existing works have investigated conflict resolution problems with various strategies considered. However, there are still several research gaps and challenges as UAS operations introduce into urban environments with larger scale and higher requirements for conflict resolution capabilities. First, existing studies have fewer considerations of the third-party risks when UAS operates in urban environments. The third-party risk is defined as the people and property not associated with and not directly benefit from the UAS operations, such as fatality risk to people on the ground [8], [9]. These risks should be considered when conducting specific conflict resolution strategies. Because existing conflict resolution methods may not perform well in a risk environment, some solution points may turn out to be high-risk areas and therefore invalid for planning. Second, conflict resolution strategies in works of literature are separately modelled and used such as only conducting scheduling or rerouting. However, some types of conflicts cannot be solved by scheduling (or speed adjustment) such as the head-to-head conflict [217]. Rerouting also has its issue. It pays a high price for downstream impact [105], as replanned routes may intersect with other initial routes causing new conflicts. What is more, there are very few studies that have combined different strategies to solve flight conflicts. A decision-making method is still lacking to adaptively select the most suitable resolution strategy for different types of flight conflicts.

This Chapter will focus on the strategic deconfliction for multiple UAVs in risk-based environments. An adaptive decision-making method is developed to solve the different types of flight conflicts with different strategies. To make the scope of this Chapter clearer, the following assumptions are made:

(1) The risk map and the pre-planned route for each UAV flight are generated by an existing study [170]. The focus of this Chapter is on strategic deconfliction among multiple UAVs.

(2) The UAV refers to both quadrotor and fixed-wing unmanned aerial vehicles. As fixed-wing UAVs are unable to hover at the speed of zero, hovering is not made an option in the problem formulation section.

6.2 Modelling of Risk-Based 4D Route and Flight Conflict

In this section, the concept of the risk-based 4D route optimization problem is introduced. Conflict detection based on the 4D routes is defined and the decision variables for solving the detected conflicts are also discussed.

The risk-based 4D route is a sequence of waypoints in discrete space with 3D coordinates and time, which considers the operational risks to third parties. The 3D coordinates include (x, y, z), while the two primary temporal information are estimated time of departure (ETD) and estimated time of arrival (ETA). The ETD is only influenced by the scheduled departure time, while the ETA can be influenced by both the ETD and flight speed. For instance, rescheduling the ETD will subsequently change the ETA, given the same flight speed. While the ETA can also be influenced if the speed is changed for a flight route, given the same ETD.

We define the 4D route on a directed graph $G = (W, A)$, where $W = \{0,1,2, \dots, w + 1\}$ is a set of waypoints. Note that waypoint0 represents the original location while waypoint $w + 1$ presents the destination. Here $A = \{A_{ij}(w_i, w_j), \forall i, j \in N_{\text{node}}\}$ is the set of arcs that connect two adjacent waypoints, where $w_i = (x_i, y_i, z_i)$ and $w_j = (x_j, y_j, z_j)$. Note that N_{node} is the total number of nodes in the graph. We further define crossing waypoint (CWP) as $W_{\text{CP}} = \{\text{CWP}_{1,} \text{CWP}_{2,} \dots, \text{CWP}_k\}, \forall \text{CWP}_k \in W$. The CWP is defined as at least two arcs intersecting with each other.

Let $T = \{t_i : \forall t_i \in [0, \infty), \forall i \in N_{\text{node}}\}$ be the set of ETAs. Note that t_i is the ETA of UAV passing the waypoint w_i. Let $U = (\{U^1, U^2, \dots, U^a, \dots, U^{N_{\text{uav}}}\}, a \in N_{\text{uav}})$ be the set of UAVs and $R_{\text{4D}} = \{R_{4D}^1, R_{4D}^2, \dots, R_{4D}^a, \dots, R_{4D}^{N_{\text{uav}}}\}$ be the set of 4D routes. Here N_{uav} is the total number of UAVs. A 4D route of U^a is denoted as $R_{4D}^a = (\{W_i^a\}, i \in N_{\text{node}})$, where $W_i^a = (\{\boldsymbol{X}_i^a, t_i^a\}, \boldsymbol{X}_i^a \in \mathbb{R}^3)$.

As shown in Fig. 6.1, four 4D routes corresponding to the respective UAVs are illustrated with different origins and destinations (ODs). An example of a 4D route of UAV U^a is denoted as $\{w_1, w_2, \text{CWP}_1, w_3, \text{CWP}_3, w_4, \text{CWP}_5, w_5\}$, which has three crossing waypoints $(\text{CWP}_1, \text{CWP}_3, \text{CWP}_5)$.

Based on the definition of the 4D route, flight conflict is defined as two or more UAVs arriving at the same waypoint while the difference of their ETAs is smaller than the minimum safe separation. Formally, a flight conflict is detected if the following condition holds

$$\begin{cases} R_{3D}^{1} \cap R_{3D}^{2}, \ldots, \cap R_{3D}^{a}, \ldots, \cap R_{3D}^{N_{\text{uav}}} = \text{CWP}_u, (a \geq 2, u \geq 1, \forall a, u \in N_{\text{uav}}) \\ \left| t_{\text{CWP}_u}^{a} - t_{\text{CWP}_u}^{b} \right| < \Delta t_{\text{conflict}}, (a, b \in N_{\text{uav}}, a \neq b) \end{cases} \tag{6.1}$$

where R_{3D}^{a} is the set of waypoints for the three-dimensional route U^a, while CWP_u is the uth crossing waypoint. Note that $t_{\text{CWP}_u}^{a}$ and $t_{\text{CWP}_u}^{b}$ are ETAs of U^a and U^b at CWP_u, respectively. Here, $\Delta t_{\text{conflict}}$ is the separation minima.

To solve the flight conflict, this Chapter proposes an adaptive decision-making method for conflict resolution to optimize pre-planned 4D routes. The known information is initial 4D routes including (x, y, z, t) at each waypoint. Our goal is to optimize the 4D routes with the objective of minimizing the number of flight conflicts, third-party risk, and operational costs. The optimization variables of the 4D route include spatial variables and temporal variables. The spatial variables are 3D coordinates (x, y, z) and they are discretized by position. The temporal variables include ETD and ETA. The resolution strategies are conducted based on these variables to solve flight conflicts.

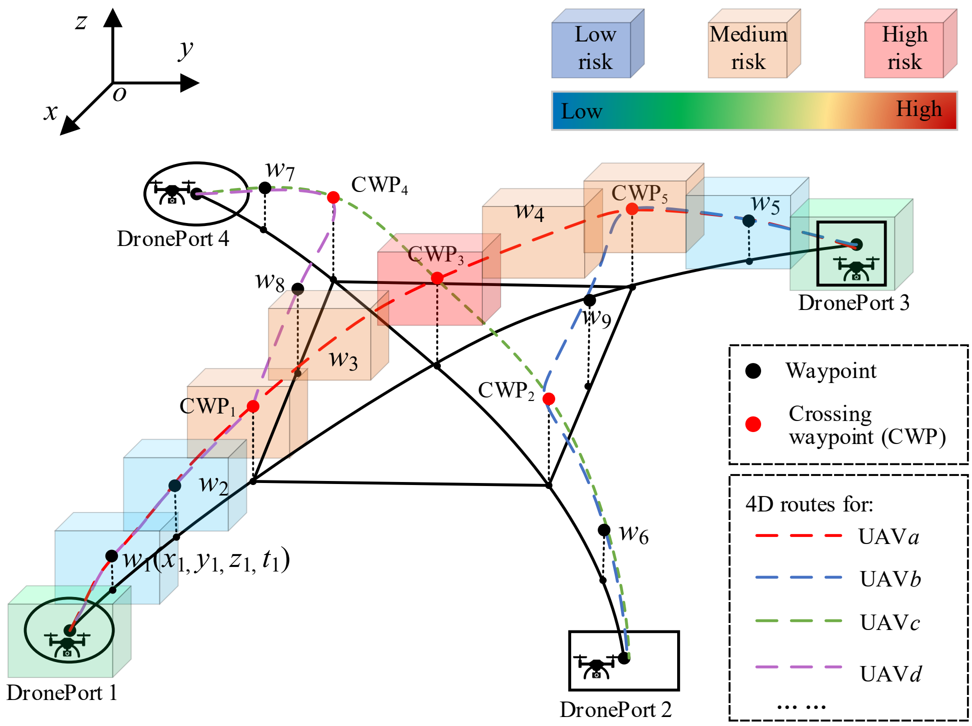


Fig. 6.1: Illustration of multi-UAV 4D routes in the risk-based airspace.

6.3.1 Strategies for Conflict Resolution

The strategies for flight conflict resolution can be classified into two aspects from temporal and spatial dimensions. From a temporal aspect, the strategies are to change the ETD and ETA. The ETD can be directly adjusted by changing the departure time of the corresponding UAV. The ETA of each waypoint has two ways to adjust. One is to change the ETD, which will change the global ETA for every single waypoint. The other is to adjust the flight speed (accelerate or decelerate) of UAV in each route segment, and the ETA of all succeeding waypoints will be altered. On the other hand, the spatial strategy is rerouting, which is to adjust the 3D coordinates of the route. The illustration of scheduling, speed adjustment, and rerouting strategies are presented in Fig. 6.2.

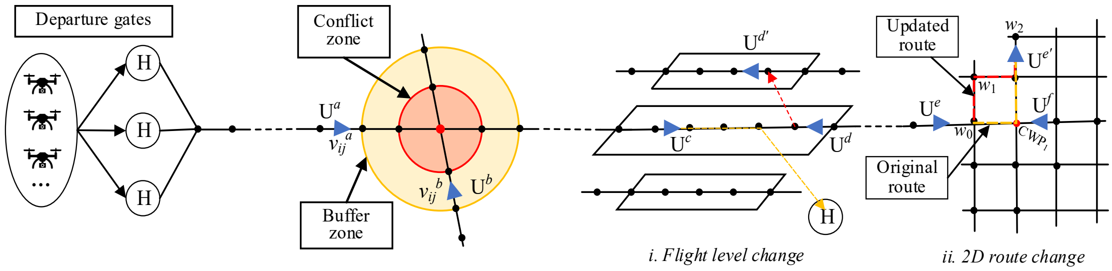
Scheduling is to adjust the estimated time of departure illustrated in Fig. 6.2(a). The ETD for a particular U^a can be presented as t^a_{Dep}. Here, $T_{\mathrm{Dep}} = \{t^a_{\mathrm{Dep}}, a \in N_{\mathrm{uav}}\}$ is the set of pre-planned departure times for all UAVs. Due to the capacity limitations of airspace, the preferred departure time proposed by UAV users may not be approved. To obtain the feasible or even optimal departure time for all missions, scheduling is required. By adjusting departure time, the ETA for each waypoint is changed accordingly, and that helps to globally resolve flight conflicts and congestion.

Speed adjustment is to solve flight conflict by increasing (decreasing) the speed, and the ETA is brought forward (backward) to avoid flight conflict from the temporal dimension. Let $V^a = \{v^a_{ij}, i, j \in N_{\mathrm{node}}\}$ be the set of speed for the 4D route of U^a, and the UAV speed at the arc A^a_{ij} is presented as v^a_{ij}. As shown in Fig. 6.2(b), U^a has speed of v^a_{ij}, and U^b has speed of v^b_{ij}, assuming $v^a_{ij} = v^b_{ij}$. The U^a and U^b are at same flight altitude and approaching the same waypoint. There is a chance that the two UAVs will have a conflict at the crossing waypoint. In this scenario, speed adjustment can be used to solve the conflict. As U^b has entered the buffer zone while U^a is outside the zone, the strategy is to increase the v^b_{ij} and decrease the v^a_{ij}, and subsequently the t^b_j at w_j is brought forward and t^a_j is brought backward. Thus, the time difference of the two UAVs is widened, and U^b can pass w_j and exit conflict zone before U^a enters.

Rerouting is another strategy for conflict resolution, and there are two ways to do so. One is only to change the flight altitude z_i and the other is to adjust the 2D coordinates (x_i, y_i) of waypoint w_i. The flight altitude change is one of the most effective strategies to solve conflict for manned and unmanned aircraft [127]. This strategy considers the remaining flight range of the flight to determine which one is climbing up or descending. For instance, a UAV approaching its destination will soon descend to land, and climbing is not considered unless there are no other solutions. As shown in Fig. 6.2(c), the 4D routes of U^c and U^d have one segment in common with the same flight altitude while they are operating in an opposite direction. In this case, scheduling and speed strategies fail to work as they can only postpone the flight conflict but cannot solve it. In that case, rerouting is the effective way and the flight altitude of U^d is changed to a higher flight level, as U^c is going to land.

Another way of rerouting is to change the 2D coordinates of the route. In certain situations, the change of 2D coordinates solves flight conflict in a more energy-efficient way as it diverts UAVs into planar space without the need for vertical movement. For instance, two UAVs operating at the same altitude have a flight conflict at a crossing waypoint. As there are a few more alternative waypoints that can be chosen to fly, the quickest strategy, in this case, is to divert one of the UAVs to avoid conflict. The principle of diverting which UAV relies on their remaining routes and destinations to maximize the smooth transiting and minimize the impact on the current traffic network. One example is given in Fig. 6.2(c). The UAVs U^e and U^f have conflict in CWP1. The U^e is heading to w_2 with original route of passing w_0, CWP1 and w_2, while U^f is moving straightly forward passing CWP1 and w_0 in sequence. In this case, we divert the U^e from original route (yellow line) to the updated one (red line), which solves the conflict and does not increase the distance-related cost.

As discussed above, different strategies have their unique advantages and weaknesses. Temporal strategies (scheduling and speed adjustment) solve flight conflict and congestion in a global view, but they are unable to solve certain types of conflicts such as head-to-head ones. Spatial strategy (rerouting) is capable of solving conflicts more locally with immediate resolution effects, but it always has a global impact on the overall flight plans. That is because the change of certain routes may influence other initial ones causing new conflicts. In summary, one strategy cannot fit all flight conflict situations. Different types of conflicts should be solved by different strategies that suit the conflicts. However, when it comes to several resolution strategies with their unique pros and cons, it is challenging to choose the most suitable one for a specific type of conflict. To meet this challenge, we propose a double-layer optimization framework, presented in the following sections.



(a) Scheduling (t) (b) Speed adjustment (v) (c) Rerouting (x, y, z)

Fig. 6.2: Decision-making strategies for UAV conflict resolutions.

6.3.2 A Decision-Making Framework for 4D Routes Optimization

In this section, a double-layer decision-making framework is proposed to optimize UAV 4D routes and the overall workflow is presented in Fig. 6.3. The first layer of the framework is constructed as an adaptive decision-making method, which is used to optimize the strategy assigned for different types of flight conflicts. The output of the first layer is the specific resolution strategies for different types of flight conflicts. The second layer of the framework is developed as a mixed-integer nonlinear programming (MINLP) model to optimize the decision variables of the strategies selected from the first layer. While an improved stochastic fractal search algorithm is used to solve the developed optimization problem. The final optimization output is the conflict-free 4D routes with a minimized number of flight conflicts, operational costs, and delays.

6.3.2.1 First layer: An Adaptive Method for Strategy Allocation

The adaptive decision-making (ADM) method is to determine which strategy should be assigned to solve what type of flight conflict. The initial multiple 4D routes are checked to detect flight conflicts, which are, if any, processed to find the best suitable resolution strategies. Let $C = \{c_1, c_2, \dots, c_i\}$ be the set of flight conflicts. Let $S = \{s_1, s_2, s_3\}$ be the set of strategies for conflict resolution. If flight conflicts are detected, the ADM method is used to select the suitable resolution strategy from S. Note that N_{fc} is the total number of conflicts detected. Here n_{s1} is the number of the scheduling strategy being allocated for conflict resolution, while n_{s2} and n_{s3} are the number of speed adjustment and rerouting strategies being used. The ith conflict is denoted as $c_i, i \in N_{\mathrm{fc}}$, and each conflict is allocated with a resolution strategy.

We further define the objective function of optimizing ith flight conflict is $f_i(\mathbf{s}_i)$, where $\mathbf{s}_i$ is a feasible solution in searching space of set S. Note that $\mathbf{s}_i = g_i(\mathbf{x}_i, t_i, v_i)$, where $\mathbf{x}_i, t_i, v_i$ are decision variables for resolution strategies. So, the objective function for conflict resolution can be denoted as $f_i(g_i(\mathbf{x}_i, t_i, v_i))$. The objective of the ADM method is to solve $f_1, f_2, \ldots, f_q$ to find better strategies $\mathbf{s}_i$. Instead of solving the flight conflicts one after another, the ADM processes all flight conflicts concurrently to obtain the best possible strategies for each type of conflict and to generate globally optimized multiple 4D routes.

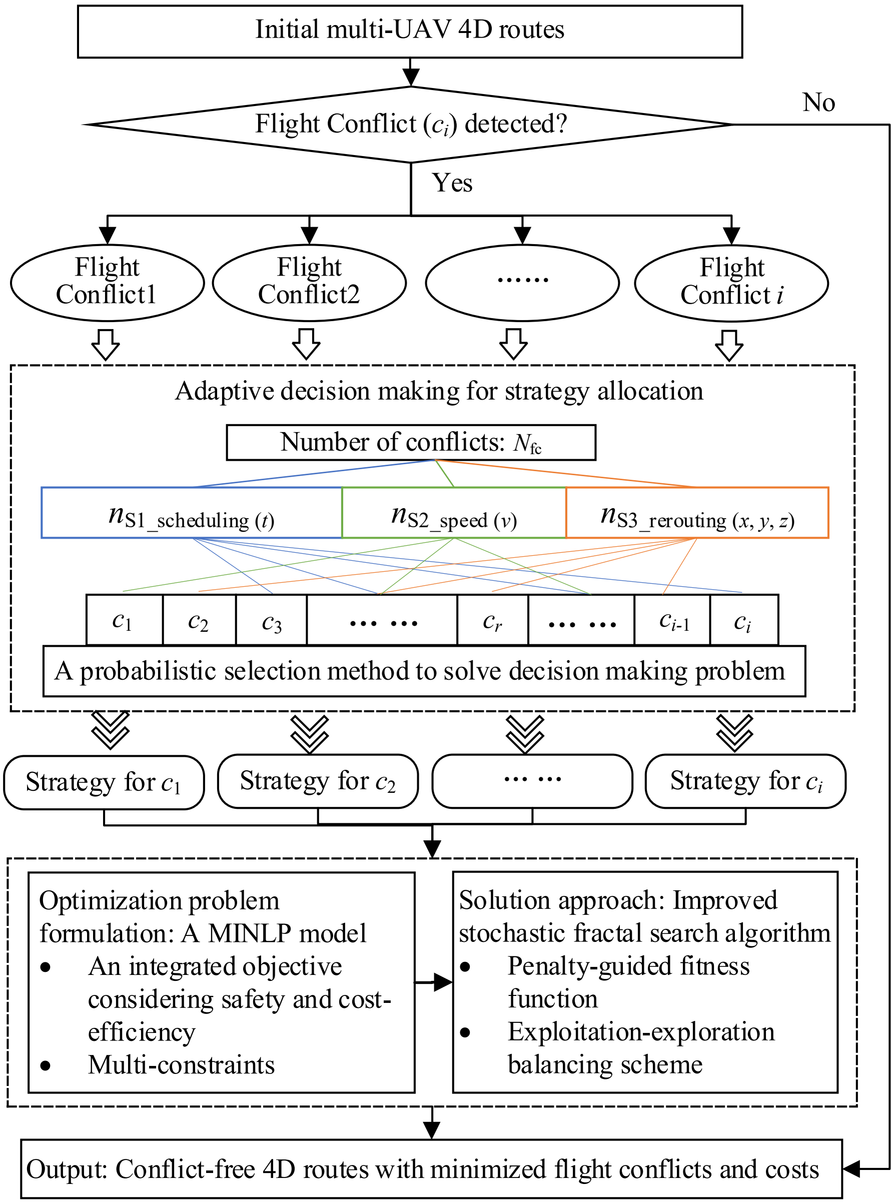


Fig. 6.3: Overall workflow of the decision-making framework.

Instead of using a combination of strategies for each UAV, a single strategy will be used for a particular UAV. For one thing, any changes on the initial flight plan (departure time, flight speed, and 3D route) will have negative impacts on either user's side or traffic management side, which should be minimized [40], [204], [205]. For another, all conflicts can be solved by a single strategy if the suitable one can be found. The ADM method is proposed to find the suitable strategy for every single UAV.

6.3.2.2 Second layer: A MINLP Model for 4D Routes Optimization

To optimize the decision variables $\mathbf{x}_i, t_i, v_i$ of the obtained strategies $s_1, s_2, \dots, s_i$, the mathematical formulation of the second layer is developed. Parameters, decision variables, objective function, and constraints used in the formulation are given as follows.

Sets

$W = \{0,1,2,\dots,w+1\}$ is the set of waypoints.

$W_{\mathrm{CP}} = \{\mathrm{CWP}_{1,} \mathrm{CWP}_{2,} \dots, \mathrm{CWP}_k\}, \forall \mathrm{CWP}_k \in W$, is the set of crossing waypoints.

$A = \{A_{ij}(w_i, w_j), \forall i, j \in N_{\mathrm{node}}\}$ is the set of arcs.

$U = \{U^1, U^2, \dots, U^a, \dots, U^{N_{\mathrm{uav}}}\}$ is the set of UAVs, and $a \in N_{\mathrm{uav}}$.

$R_{\mathrm{4D}} = \{R_{4D}^1, R_{4D}^2, \dots, R_{4D}^a, \dots, R_{4D}^{N_{\mathrm{uav}}}\}$ is the set of 4D routes.

$V^a = \{v_{ij}^a\}$ is the set of speed for the 4D route of U^a.

$T_{\mathrm{Dep}} = \{t_{\mathrm{Dep}}\}$ is the set of pre-planned departure times for all UAVs.

$T_0 = \{t_0\}$ is the set of actual departure times for all UAVs.

$T_i^a = \{t_i^a : \forall t_i^a \in [0, \infty), \forall i \in N_{\mathrm{node}}, a \in N_{\mathrm{uav}}\}$ is the set of ETAs of U^a passing w_i.

Parameters

$w_i(x_i, y_i, z_i)$: coordinates of waypoint i, $\forall i \in N_{\mathrm{node}}$.

$A_{ij}(w_i, w_j)$: arc connects two adjacent waypoints w_i and w_j.

r_i: risk index value of waypoint w_i.

U^a: ath UAV.

CWP_k: kth crossing waypoint. Here $k \in K$, and K is the number of crossing waypoints.

v_{ij}^a: speed of ath UAV at the arc A_{ij}.

t_{Dep}^a: pre-planned departure time of U^a.

t_0^a: actual departure time of U^a.

$t_{w_i}^a$: ETA of U^a passing waypoint w_i.

R_{4D}^a: 4D route of U^a. $R_{4D}^a = (\{W_i^a\}, i \in N_{\mathrm{node}}, a \in N_{\mathrm{uav}})$.

Decision variables

$w_i^a \in \{0,1\}$: equals to 1 if U^a passes waypoint w_i; equals to 0 otherwise.

v_{ij}^a: the speed of U^a passes arc A_{ij}, which is taken as the integer variable with the unit of km. Here $v_{\min}^a \le v_{ij}^a \le v_{\max}^a$.

t_0^a: the actual departure time of U^a, which is taken as the integer variable with the unit of second.

Objective function (MINLP)

The objective of the proposed model is to minimize the third-party risks, flight delays, and airborne time-related costs. The overall objective function is denoted as

$$\min: f_{\text{obj}} = \omega_{\text{risk}} R_{\text{TP}} + \omega_{\text{t\_delay}} T_{\text{delay}} + \omega_{\text{t\_air}} T_{\text{air}} \tag{6.2}$$

where ω_{risk}, $\omega_{\text{t\_delay}}$, and $\omega_{\text{t\_air}}$ are weightage factors for risk, delay, and airborne time, respectively. Note that $\{\omega_{\text{risk}}, \omega_{\text{t\_delay}}, \omega_{\text{t\_air}}\} \in [0,1]$, and $\omega_{\text{risk}} + \omega_{\text{t\_delay}} + \omega_{\text{t\_air}} = 1$.

Safety is the top priority for UAV operations in urban environments. The first sub-objective of the model is to minimize the third-party risks caused by UAVs, presented as

$$R_{\text{TP}} = \sum_{a=1}^{N_{\text{uav}}} \sum_{i \in \text{W}_\text{p}} r_i^a w_i^a \tag{6.3}$$

where R_{TP} is the total risk of UAV routes, and r_i^a is the risk value of waypoint w_i^a of U^a. Note that W_p is the set of waypoints that are passed by UAV.

The second sub-objective of the model is to minimize the overall flight delays

$$T_{\text{delay}} = \sum_{a=1}^{N_{\text{uav}}} |t_0^a - t_{\text{Dep}}^a| \tag{6.4}$$

where T_{delay} is the total delay of all UAVs. Here t_0^a is the actual departure time and t_{Dep}^a is the pre-planned departure time of UAV U^a. Note that the T_{delay} is a nonlinear function, as it is the summation of the absolute value of the time difference between actual departure and pre-planned departure,

Compared with flight distance, airborne time is more crucial for UAV operations in terms of operational safety (limited battery life) and efficiency (time-related flight costs). The third sub-objective is to minimize the total airborne time, denoted as

$$
\begin{aligned}
T_{\text{air}} &= \sum_{a=1}^{N_{\text{uav}}} \sum_{w_i,\, w_j \in \mathrm{W_r}} \left(\frac{d_{ij}}{v_{ij}^{a}}\right) \\
&= \sum_{a=1}^{N_{\text{uav}}} \sum_{w_i,\, w_j \in \mathrm{W_r}} \left(\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}/v_{ij}^{a}\right)
\end{aligned} \tag{6.5}
$$

in which T_{air} is the total airborne time consumed by all UAVs, and the total number of UAVs is denoted as N_{uav}. Here, w_i and w_j are neighboring waypoints, which belong to a set of waypoints $\mathrm{W_r}$ for a route. The neighboring correlation is defined by the absolute difference of their index i and j, denoted as: $|i - j| = 1$. Note that d_{ij} is the Euclidean distance of the arc $A_{ij}(w_i,\ w_j)$, and v_{ij}^{a} is the speed of U^a in A_{ij}. The function of T_{air} is nonlinear, as the decision variable v_{ij}^{a} is the divisor in the function. Both Eq. (6.4) and Eq. (6.5) are nonlinear functions, therefore the optimization problem is developed as an MINLP model.

Constraints

$$
\begin{cases}
x_{i+1} = x_i + \xi_x^i, x_i \in \{0, l, 2l, \dots, X_{\max}\} \\
y_{i+1} = y_i + \xi_y^i, y_i \in \{0, l, 2l, \dots, Y_{\max}\} \\
z_{i+1} = z_i + \xi_z^i, z_i \in \{0, l, 2l, \dots, Z_{\max}\}
\end{cases}, \forall\, \xi_x^i, \xi_y^i, \xi_z^i \in \{-l, 0, l\}, \forall i \in N_{\text{node}} \tag{6.6}
$$

$$
|\xi_x^i| + |\xi_y^i| + |\xi_z^i| \neq 0, \forall\, \xi_x^i, \xi_y^i, \xi_z^i \in \{-l, 0, l\} \tag{6.7}
$$

$$
|(t_0^a + T_{\text{air}}^a) - (t_{\text{Dep}}^a + \mathrm{T_{air\_planned}})| \leq T_{\text{delay}}^a, \forall a \in N_{\text{uav}} \tag{6.8}
$$

$$
\sum_{i,j \in N_{\text{p}}} \left(\frac{d_{ij}}{v_{ij}^{a}}\right) \leq T_{\text{BTRY}}^a, \forall a \in N_{\text{uav}} \tag{6.9}
$$

$$
\sum_{a \in N_{\text{uav}}} U_{\text{O}}^a = \sum_{b \in N_{\text{uav}}} U_{\text{D}}^b \tag{6.10}
$$

$$
v_{\min}^a \leq v_{ij}^a \leq v_{\max}^a, \forall a \in N_{\text{uav}} \tag{6.11}
$$

$$
0 < t_{\text{Dep}}^a \leq T_{\text{slot}}, t_{ij}^a, t_0^a > 0, \forall i,j \in N_{\text{node}}, \forall a \in N_{\text{uav}} \tag{6.12}
$$

Constraint (6.6) ensures the consistency of flight motion in discrete space, and 26 candidate waypoints can be chosen as the next waypoint, including planar diagonal and cubical diagonal. Constraint (6.7) restricts that hovering is not allowed during flight, as the UAV refers to both the quadrotor and the fixed-wing unmanned aerial vehicles in this Chapter. Constraint (6.8) restricts the arrival delay of the flight. The flight arrival time to a destination should not exceed the threshold T_{delay}^{a} , otherwise the corresponding flight will be regarded as a delayed flight. Where t_0^a is the actual departure time and T_{air}^{a} is the actual airborne time, while t_{Dep}^{a} is pre-planned departure and $T_{\text{air\_planned}}$ is the pre-planned airborne time. Constraint (6.9) ensures the flight time of the planned route does not exceed the battery life of the UAV. Constraint (6.10) ensures that traffic flow from origins to destinations is balanced. If a flight cannot reach its destination, that flight will be rejected. Constraint (6.11) ensures the assigned flight speed is within the range of UAV speed capability. Constraint (6.12) ensures the assigned departure time for each flight is within the given departure time window.

6.3 Improved Stochastic Fractal Search (ISFS) Algorithm

In Section 6.3, the adaptive conflict resolution problem is developed as an MINLP optimization model. The model has several decision variables that act under various constraints. The bounds of the constraints include both simple forms such as the range of flight speed (Constraint (6.11)) and nonlinear relationships such as the flight duration (Constraint (6.9)). The MINLP has been proven as a Non-Deterministic Polynomial (NP-hard) problem [218] with the complexity of $O(n!)$ [219]. With the increase of the problem size n, the search space for the high-dimensional optimization problem increases dramatically. The solution quality and computational efficiency for such type of problem become more unacceptable if using classical optimization methods such as exhaustive methods [211]. Furthermore, as the conflict resolution problem is highly combinatorial, the classical methods are only applicable to solve the problem with a small number of aircraft [211], [220]. That motivates the use of metaheuristic algorithms.

Metaheuristic algorithms, as approximate methods, are not sensitive to the size of the search space [221], [222] and have become the primary alternative to solve large-scale global optimization problems. The stochastic fractal search (SFS) algorithm is one of the novel and powerful metaheuristic algorithms for solving both discreet and continuous optimization problems [194]. The SFS is a stochastic algorithm, and it performs well in global optimization problems. That makes it a suitable algorithm to solve the proposed constrained CD&R optimization problem with objectives to globally minimize the total number of flight conflicts and total costs. Besides, the SFS algorithm has been proven the better performance in terms of solution quality and convergence time in various engineering and optimization problems [194], [223], [224], which enables the strategic deconfliction to be completed in a shorter time with a better solution. That subsequently improves the UAV traffic performance and accelerates the process for strategic flight planning. The flexibility of the base SFS algorithm also allows it to be extended and implemented in solving the proposed MINLP problem in this Chapter.

The SFS algorithm has two key processes, the diffusion process and the update process, to provide exploitation and exploration capabilities in solving optimization problems. The fundamental concept of SFS is to mimic the diffusion property in fractals, which can be widely seen in the self-similarity of patterns in nature such as trees, crystals, snowflakes, etc. Based on the diffusion property, each particle (point or individual solution) tries to simulate the branching property of a dielectric breakdown and propagate from the central point. This property allows the central point to create a series of new particles adjacent to each other with the capability of moving in different directions at various distances. With the diffusion property, the initial energy at a central point can be effectively exploited to other objects that have reactions to this energy. The exploration of solutions is performed in the selection and update process based on the fitness function.

In this Chapter, we improve the SFS algorithm from two aspects. First, to minimize the number of flight conflicts, we propose a conflict penalty-guided fitness function to evaluate the solution candidates. Solutions with more flight conflicts will have a higher probability of being discarded in the next optimization process. After optimization, the obtained solution achieves conflict-free routes with a minimum number of flight delays and total cost. Second, we introduce an exploitation and exploration balancing scheme to improve the effectiveness of conflict resolution methods in a risk-based environment. The scheme improves the searching diversity of the SFS algorithm by considering the fitness value and solution point distance to the best point. That prevents new solution points from being too close to the existing best point, which avoids the solutions being trapped in local minima. With the improved SFS, we develop adaptive decision-making method to allocate the best suitable strategies for different types of flight conflicts.

6.4.1 Penalty-Guided Fitness Function in Exploitation and Exploration Processes

6.4.1.1 Exploitation and exploration processes

In the SFS algorithm, the diffusion process performs exploitation tasks while the update process provides exploration capabilities. The diffusion process is to search for new solutions by creating particles adjacent to the central point. The position of new particles in the search space is distributed based on Gaussian random walks, denoted as

$$P_i^{\eta} = P_i + \mathrm{Gaussian}(\mu_P, \sigma) = P_i + \frac{1}{\sqrt{2\pi}\sigma} e^{-(\mu_P-\sigma)^2/2\sigma^2} \tag{6.13}$$

$$P_i^{\eta} = P_i + \mathrm{Gaussian}(\mu_P, \sigma) + (\varepsilon P_{\mathrm{best}} - \varepsilon' P_i) \tag{6.14}$$

where P_i^{η} is the ηth new particle created by central particle P_i, and η is the number of new particles generated. Here μ_P is the mean of Gaussian distribution of new particle positions, which is equal to the absolute value of ith solution particle denoted as $|P_i|$. Here, P_{best} is the particle with the best solution in the current group. Note that ε and ε' are weightage factors to adjust the degrees of exploitation and exploration. They are random values following the uniform distribution and are in the range of [0,1]. Here, σ is the standard deviation, computed by

$$\sigma = \left| \frac{\log\left(N_{\mathrm{Iter}}\right)}{N_{\mathrm{Iter}}} P_i - P_{\mathrm{best}} \right| \tag{6.15}$$

Note that, N_{Iter} is the number of iterations.

After the diffusion process, the particles are evaluated by the fitness function. The update process is to select particles that have a high fitness value while ensuring the high diversity of solution particles. The first update process is to select the particles with good fitness values. The jth diffused seed particle of P_i will be updated, only if the following condition holds

$$\begin{cases} P_{a_i} < \varepsilon, \varepsilon \in [0,1] \\ P_{a_i} = \dfrac{\mathrm{rank}(P_i)}{n_{\mathrm{p}}} \end{cases} \tag{6.16}$$

in which P_{a_i} is the probability of a particle being updated, and n_{p} is the number of particles. Particles with good solutions are ranked higher in the diffusion group and are given a smaller value of $\mathrm{rank}(P_i)$, so that these particles are secured and exploited. On the contrary, particles with bad solutions will be updated to search for new positions.

The first update process function is denoted as

$$P_i'(j) = P_r(j) - \varepsilon(P_t(j) - P_i(j)) \tag{6.17}$$

where $P_i'(j)$ is the updated position of the ith particle, and j is the dimension of current points. Here, P_r and P_t are other particles randomly selected from the current population.

The second update process is to ensure high diversity of solutions by considering the positions of other solution particles in the whole population, formulated as

$$P_i'' = P_i' - \varepsilon'(P_t' - P_{\text{best}}), \varepsilon' \leq 0.5 \tag{6.18}$$

$$P_i'' = P_i' + \varepsilon'(P_t' - P_r'), \varepsilon' > 0.5 \tag{6.19}$$

Three reference points are employed in the update process, and they are the position of the current particle P_i', randomly selected particle P_t' (or P_r') and the position with the best particle P_{best}.

6.4.1.2 Penalty-guided fitness function

Once obtained the initial population of particles, we evaluate the fitness value and select the best individual solution. In this Chapter, the objective function is to minimize the total operational cost that includes third-party risks, flight delay, and airborne time (as denoted in Eq. (6.2)). What is more, flight conflict is the most important risk factor, and it should be minimized in the strategic deconfliction phase. As the flight conflict may dominate the objective function, it is considered a significant penalty in the fitness function of the algorithm. Solution candidates who have flight conflicts will be significantly penalized with an extra value, and the overall fitness function is denoted as

$$f_{\text{value}} = f_{\text{obj}}(w, t, v) + \psi(w, t, v) \tag{6.20}$$

where $\psi(w, t, v)$ is the penalty function. The total conflict penalty is correlated with the number of conflicts, which can be computed by

$$\psi(w, t, v) = \sum_{i=1}^{N_{\text{fc}}} \omega_{\text{fc}} f_{\text{obj}}(w, t, v)_i \tag{6.21}$$

in which ω_{fc} is the weightage factor of flight conflict and N_{fc} is the total number of flight conflicts in a solution.

Assuming an optimization problem has a solution with dimension N, each individual particle is considered a potential solution, and it is an N-dimensional vector. The initialization of particles is randomly conducted within the constraints of the problem, which is denoted as

$$P_i = B_{\text{lower}} + \gamma\left(B_{\text{upper}} - B_{\text{lower}}\right) \tag{6.22}$$

where P_i is the ith root particle in the population. Here B_{lower} and B_{upper} are the lower bound and upper bound of the constraint vectors. Note that γ is a random number and $\gamma \in [0,1]$. When $\gamma = 0$, the ith root particle $P_i = B_{\text{lower}}$, while $P_i = B_{\text{upper}}$ when $\gamma = 1$

6.4.2 Exploitation-Exploration Balancing Scheme

Exploitation and exploration are two key requirements for optimization algorithms, as exploitation is to utilize the current best solution while exploration is to search for better solutions [225]. A good trade-off between them yields a high-quality solution while maintaining a reasonable computational time [226]. In the SFS algorithm, the update processes do not consider the distance between the solution candidate to the best particle in the population. Too many particles close to the best one leads to an ineffective diversity of exploration.

To solve this problem in meta-heuristic search algorithms, researchers proposed a fitness-distance balance (FDB) method [227]. The FDB method proposed a score value to evaluate each particle in selection and update processes. The score value incorporates the fitness value of each particle (computed by Eq. (6.20)) and the distance value between the solution particle and the best particle. That ensures particles with high fitness values are selected while avoiding selecting the particles that are too close to the best particle.

The score values of each particle in the population are computed, and the score vector S_P is presented as

$$S_P = [s_1 \quad s_2 \quad \cdots \quad s_n]^T \tag{6.23}$$

where s_1 is the score value of P_1, and n is the number of particles in the population.

To calculate the distance values between solution particles and the best particle, we denote the position of the best particle in the population is X_{best}. The distance value for each solution particle is computed by

$$d_{P_i} = \sqrt{(x_{1P_i} - x_{1P_{\text{best}}}) + (x_{2P_i} - x_{2P_{\text{best}}}) + \cdots + (x_{\zeta P_i} - x_{\zeta P_{\text{best}}})} \tag{6.24}$$

where d_{P_i} is the distance between the ith solution position $(x_{1P_i}, x_{2P_i}, \dots, x_{\zeta P_i})$ of P_i ($P_i \neq P_{\text{best}}$) and the best solution position $(x_{1P_{\text{best}}}, x_{2P_{\text{best}}}, \dots, x_{\zeta P_{\text{best}}})$ of P_{best}. Here, ζ is the dimension space of the particles ($\forall_{i=1}^{n}, P_i \in \mathbb{R}^{\zeta}$).

Accordingly, we have the distance vector D_P, presented as

$$D_P = [d_1 \quad d_2 \quad \cdots \quad d_n]^T \tag{6.25}$$

The fitness value is computed by Eq. (6.20), and the fitness vector F_P is expressed as

$$F_P = [f_1 \quad f_2 \quad \cdots \quad f_n]^T \tag{6.26}$$

To prevent the dominance of fitness value or distance value in the score function, normalizations are conducted. Specifically, the distance vector and the fitness vector are divided by the maximum value among their vectors, respectively. The normalized distance vector is denoted as $D_{P\_\text{norm}} = D_P/d_{\max}$ ($d_{\max} = \max\{d_1, d_2, \dots, d_n\}$ and $d_{\max} \neq 0$). While the normalized fitness vector can be denoted as $F_{P\_\text{norm}} = F_P/f_{\max}$ ($f_{\max} = \max\{f_1, f_2, \dots, f_n\}$ and $f_{\max} \neq 0$). After the normalizations, all distance values and fitness values fall within the range of [0, 1]. The obtained score vector S_P is then denoted as

$$S_P = \omega_{\text{FDB}} F_{P_{\text{norm}}} + (1 - \omega_{\text{FDB}}) D_{P_{\text{norm}}} \tag{6.27}$$

where ω_{FDB} is a parameter to adjust the weightage of fitness value and distance value when conducting the selection and update process in the FDB method, which is in the range of [0, 1]. When $\omega_{\text{FDB}} = 1$, only the fitness value will involve and the solution searching process is conducted in a greedy manner, meaning that the exploitation is effective. In contrast, when $\omega_{\text{FDB}} = 0$, only the distance value will be counted, which provides a good variation of population and great exploration capability.

The pseudo-code of the ISFS algorithm is presented in Algorithm 1. The input includes all initial 4D route information and the Risk Cost dataset, which is generated by computing the risk index in pertaining environments [170]. The ISFS algorithm has two key processes. One is the diffusion process (Line 5 to Line 8), which is to exploit the first obtained best solution by searching its neighbors. The other is the update process (Line 9 to Line 16). The first update process is to randomly explore the solutions in the population, while the second update process is to ensure the diversity of the search by considering other solution points in the population.

Throughout the whole process, the fitness value is computed by the improved penalty-guided fitness function. That guides the solution to a minimum number of flight conflicts. On the other hand, the improved fitness-distance balancing strategies are used in the update process (Line 11 and Line 14) to avoid the explored points that are too close to the current best point, which improves the diversity of ISFS and avoids the search being trapped in local minima. The ISFS algorithm is an evolutionary metaheuristic algorithm. As shown in Algorithm 1, the number of executions for each script in the main loop of the ISFS is n. That gives the complexity of ISFS $O(n)$.

6.4.3 ISFS-Based ADM Method for Strategy Allocation

Based on the ADM framework and the ISFS algorithm, we develop the ADM method using a probabilistic scheme to optimize the strategies allocated for conflict resolution. The pseudo-code of the ADM method is presented in Algorithm 2.

The core of this method is to sample explicit probabilistic models of the promising solutions, which guides the search for the optimum. The input of the ADM method is the lower bound and upper bound of decision variables. The initialization is to generate a probability matrix for strategy allocation. The main optimization loop (Line 3 to Line 19) is an incremental process of the probability that a strategy with a better fitness value will be chosen for conflicts. In each iteration, the dominant population with a better fitness value is selected and the corresponding probability is updated by the formula shown in Line 18. The learning rate l_{rate} is introduced to adjust the trade-off between the exploitation of the last-step probability matrix and the exploration for the next probability increment. Here the l_{rate} is taken as 0.5 to balance exploitation and exploration. For the ADM algorithm, the complexity of the outer loop is $O(n)$ while the ISFS algorithm is called in the inner loop. That makes the complexity of the overall ADM algorithm $O(n^2)$.

Algorithm 1: Improved Stochastic Fractal Search (ISFS) Algorithm

1: Input: Initial 4D routes *(x, y, z, t),* RiskCost Dataset
2: Initialization: generate a population (P) of n solution points with N-dimension
3: Selection: obtain the first best solution by evaluate the fitness value
4: **while** $i <= n_{\text{gen}}$ **Do** % Optimization loop, n_{gen} is the number of generation loops
5:   **begin** Diffusion process (Exploitation)
6:   Searching neighbors of the first best solution point (**Eq. (6.13)** and **Eq. (6.14)**)
7:   Update the fitness value and the best solution point
8:   **end**
9:   **begin** Update process (Exploration)
10:   *First update process*
11:   Update the position of solution based on two random points in P (**Eq. (6.17)**) and fitness-distance balancing (**Eq. (6.27)**)
12:   Update the fitness value and the best solution point
13:   *Second update process*
14:   Update the position of solution based on other solution points in P (**Eq. (6.18)** and **Eq. (6.19)**) and the fitness-distance balancing (**Eq. (6.27)**)
15:   Update the fitness value and the best solution point
16:   **end**
17: **end**
18: **Return** (*BestSolution*, *fitnessValue*)

Algorithm 2: Adaptive Decision Making (ADM) Algorithm for Strategy Allocation

```
Input: lower bounds and upper bounds of (x, y, z, t, v)
Initialization: probability update function P_update of strategy allocation
while i<= n_gen Do % main loop
    Generate species: Randomly allocate the strategies (Sj) for each conflict
    for j = 1:3 Do
        if j==1 Do % use S1: Scheduling strategy
            Solution1=ISFS(species, P_update);
            else if j==2 Do % use S2: Speed adjustment strategy
                Solution2=ISFS(species, P_update);
            else Do % use S3: Rerouting strategy
                Solution3=ISFS(species, P_update);
            end if
        end if
    end for
    FitnessValue=FitnessFunction(Solution1, Solution2, Solution3)
    [FitnessValue, index] = sort(FitnessValue);
    dominantPop{:, 1} = species{index(:), 1}; %select dominant population
    P_update = (1-l_rate)*P_update+l_rate*dominantPop/dominantNo; %update probability matrix
end while
Return (BestSolution, FitnessValue)
```

6.4 Numerical Studies and Results

Simulations are conducted in a real-world urban environment to demonstrate the performance of the proposed adaptive decision-making framework. A comprehensive sensitivity analysis is performed for the multiple parameters used in the proposed model. The proposed algorithm is also applied to an increasing density of air traffic and algorithm comparisons are carried out to demonstrate the effectiveness and reliability of the ISFS algorithm.

6.5.1 Environment Setup and Initial 4D Routes Generation

A typical urban environment is selected for third-party risk modelling and risk-based route planning and decision making. The size of the selected low-altitude airspace is 6,000m×6,000m×120m (width, length, and altitude). Based on the AirMatrix concept [55], the airspace is divided into standard discrete air blocks with a size of 100m×100m×30m. The selected urban area is full of dense high-rise buildings, central business districts, city squares, shopping centers, residential areas, parks, etc., which are representative of modern megacities. More detailed environmental risk factors and data, such as population density and vehicle density, can be found in a previous study [170].

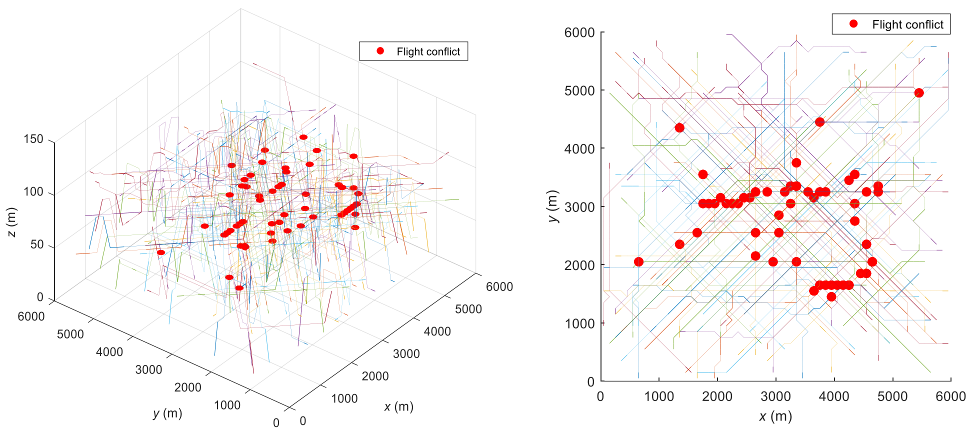
With the risk mapping data, we perform risk-based 4D route planning for all UAVs. We assigned 100 flights for this simulation in the selected airspace. The origin point and destination point of each mission are randomly selected within the boundaries, and the flight distance of each assigned mission is in the range of [2, 8] km. The time window for the departure of the 100 flights is 60 minutes, and the ETD for each flight is randomly assigned within the range. The default speed of the UAV is taken as 8 m/s. We use an exact method, the Dijkstra algorithm, to obtain the optimal pre-planned 4D routes.

Other parameters used in this study are given in Table 6.1. There are five decision variables (x, y, z, t, v) for each 4D route. Based on the boundary of the simulation environment and air block size, the range of (x, y, z) can be determined. The range of UAV speed is taken as [5, 15] m/s based on one of the most used drone specifications [228]. The time window for departure is taken as 60 minutes, so the range of ETD for 100 flights is [1, 3600] s. There are three constraint thresholds. The safe separation for flight conflict is determined by the minimum time difference of consecutive UAVs passing the same waypoint. The succeeding UAV should not reach the conflict zone boundary before the preceding UAV exists in the zone (illustrated in Fig. 6.2(b)). Based on that, the separation minima $\Delta t_{\text{conflict}}$ for flight conflict can be obtained as 30 seconds. The threshold of flight delay T_{delay}^{a} is taken as 20 minutes (1200 s), and the battery duration T_{BTRY}^{a} is taken as 25 minutes (1500 s). As flight missions may need to be taken over before the battery of the previous flight runs out, a slot of 5 minutes (300 s) should be allocated for the takeover. As to the weightage factors of the three sub-objectives in the MINLP model, we take the weightage of operational risk ω_{risk} is 0.5, while the flight delay weightage $\omega_{\text{t\_delay}}$ and airborne time weightage $\omega_{\text{t\_air}}$ are taken as 0.25. Here, we take the conflict penalty weightage ω_{fc} as 0.10 times of total cost index in each corresponding iteration.

Table 6.1: Parameters used in simulation studies.

Decision variables	Range of value	Constraint thresholds	Value	Weightage factors	Value
x	[1, 60]	$\Delta t_{\text{conflict}}$ (s)	30	ω_{risk}	0.50
y	[1, 60]	T_{delay}^{a} (s)	1200	$\omega_{\text{t\_delay}}$	0.25
z	[1, 4]	T_{BTRY}^{a} (s)	1500	$\omega_{\text{t\_air}}$	0.25
v (m/s)	[5, 15]			ω_{fc}	0.10
ETD (s)	[1, 3600]				

With the environment set up, we generate 100 UAV 4D routes and detect the flight conflicts as shown in Fig. 6.4. The OD pairs of 100 flights are randomly distributed in the environment map, and 74 flight conflicts are detected. Basically, there are two characteristics of conflict distribution. The most obvious one is in the middle of the simulation map where flight routes densely intersect with each other, and the dense traffic causes more conflicts than other areas. The conflict also occurs at waypoints where more than two routes are intersecting with. For instance, as shown in Fig. 6.4(b), the conflicts in positions of (1500, 4300), (3900, 4400), and (5500, 5000) are not in a dense area, but they have three routes crossed at the positions, which makes these waypoints vulnerable to have flight conflicts. All these potential flight conflicts threaten the safe operations of UAVs in urban areas and need to be mitigated.



(a) 3D view (b) Top view

Fig. 6.4: Pre-planned 4D routes with detected flight conflicts (74 conflicts).

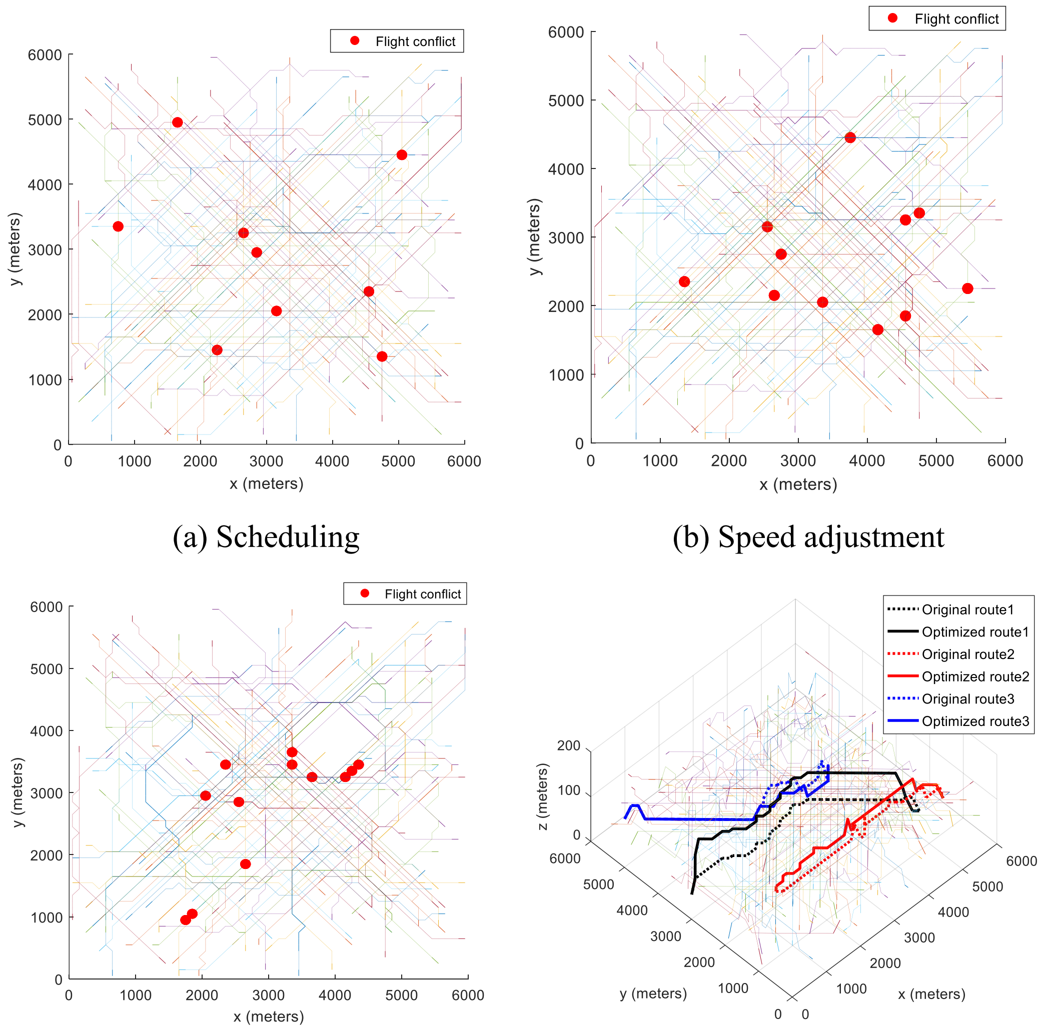
6.5.2 Results of Conflict Resolution and 4D Routes Optimization

We use our proposed adaptive decision-making method to optimize the obtained pre-planned 4D routes. The simulations are conducted in four groups by using different strategies: (1) Scheduling, which means only the estimated time of departure (ETD) for UAV flight t_0^a is the decision variable to optimize the routes using the proposed MINLP model. (2) Speed adjustment, only the flight speed v_{ij}^a is the decision variable. (3) Rerouting strategy, only the flight altitude and 2D coordinates can be changed. (4) Adaptive decision-making method, which adaptively employs all three individual strategies as presented above. The decision variables of the ADM strategy are (w_i, v_{ij}^a, t_0^a), and the MINLP model is used to optimize the 4D routes. The ISFS algorithm is used to solve the developed optimization problems, and the ADM method is used to find the optimal solution for strategy allocation. The total cost index computed by Eq. (6.20) is normalized to [1, 100,000] for all four simulations by using four different strategies. The obtained results are presented in Fig. 6.5. and

Table 6.2. The number of each strategy used in the ADM method is presented in Fig. 6.6.

After optimization, flight conflicts are significantly reduced in all situations where different strategies are used. In the temporal dimension, scheduling and speed adjustment strategies perform well with a reduction of conflict number from 74 (Fig. 6.4(b)) to 9 (Fig. 6.5(a)) and 11 (Fig. 6.5(b)), respectively. These two strategies do not change the 3D routes, which has less impact on global flight plans. The conflict distributions are also not congested, which gives more space for further resolution reactions. As to the rerouting strategy, it changes the overall layout of the pre-planned routes. The flight conflicts in optimized routes are reduced from 74 to 12, whereas the conflicts are congested in certain areas, such as the position of (4000, 3000) in Fig. 6.5(c). Our proposed ADM method performs the best, with no flight conflict exists in the optimized routes as shown in Fig. 6.5(d).

The locations of the conflicts are not the same in the three optimized routes by individual strategies. That is because each of the strategies has its unique way to optimize the 4D routes. By adjusting the departure time, the scheduling strategy globally changes the ETA of all waypoints for the pertaining route, and that avoids the traffic getting congested at one area in a short period of time. It subsequently avoids conflicts, especially conflict clusters. That explains the sparse conflict distribution in Fig. 6.5(a). The speed adjustment optimizes the 4D routes by changing the flight speed of UAV. Same as scheduling, it also does not change the layout of the routes, which avoids conflict clusters by allocating different speeds for conflict affected UAVs. For the rerouting strategy, it optimizes the 4D routes in spatial dimension by diverting the flight to avoid conflicts. However, rerouting increases the risk cost, as any change to the pre-planned optimal routes will increase the risk cost index, and there are still conflict clusters as shown in Fig. 6.5(c). On the other hand, our proposed ADM strategy adaptively employs scheduling, speed adjustment and rerouting strategies to select the best suitable strategy for different types of conflicts. For instance, to solve head-to-head flight conflicts, a rerouting strategy is used. As shown in Fig. 6.5(d), three routes are optimized by rerouting strategy, as the locations of these routes have several conflict clusters and head-to-head conflicts (Fig. 6.4(b)). The affected routes are rerouted to solve these clusters. Specifically, the flight altitude of the route1 is changed and the 2D coordinates (x, y) of the route2 are completely changed, while the route3 is locally diverted to avoid flight conflicts.



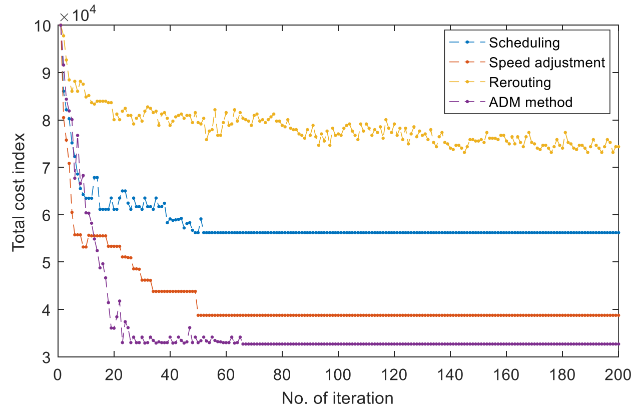
(a) Scheduling (b) Speed adjustment

(c) Rerouting (d) Adaptive dicision-making

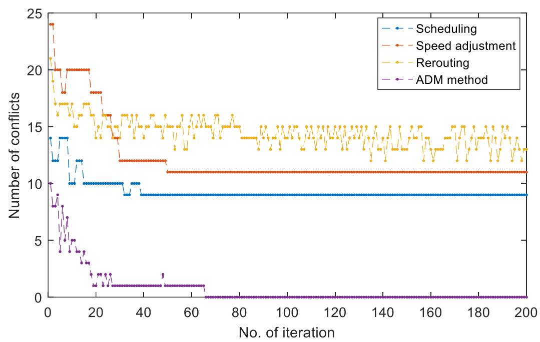
Fig. 6.5: Optimization results by using different conflict resolution strategies.

The performance of optimized 4D routes is evaluated by cost index, number of flight conflicts and number of flight delays, which are presented in Fig. 6.6(a), Fig. 6.6(b) and Fig. 6.6(c), respectively. As discussed above, the proposed ADM method takes advantage of the other three individual strategies to provide the most suitable solution for each type of conflict. Its performance excels among the other three strategies, with the lowest cost index of 32697.21 and no flight conflict, shown in

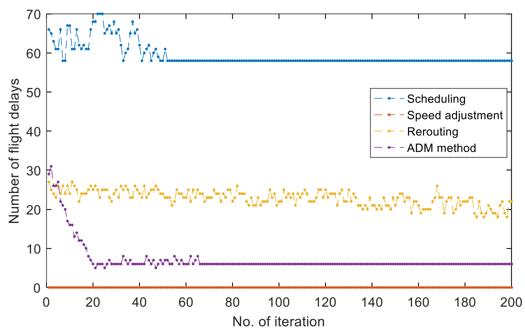
Table 6.2, meaning that no flight is rejected due to conflicts. Followed by the speed adjustment strategy, with a cost index of 38764.99. And it performs the best in reducing the number of flight delays, as it does not change the departure time and flight route. So, the arrival delay is controllable within the threshold of T^{a}_{delay}, though the number of flight conflicts is large.



(a) Total cost index



(b) Number of flight conflicts



(c) Number of flight delays

Fig. 6.6: Comparison of different conflict resolution strategies.

As to the scheduling strategy, it has the second least conflict number, but it causes the most flight delays as shown in

Table 6.2. The reason is that the scheduling strategy optimizes the 4D routes only by adjusting the departure time, which most of the time postpones departure time and subsequently causes flight delays. Lastly, for rerouting strategy, it has the worst cost index and a huge number of conflicts. As the pre-planned 3D routes are generated by the Dijkstra algorithm, which is optimal. Any change in the pre-planned routes will increase the risk cost and flight distance, which are two key contributors to compute the fitness value. What is more, the change of pre-planned routes resolves some conflicts, however, it also has an impact on other unchanged routes, which may produce new flight conflicts.

Speaking of convergency speed, the scheduling and speed strategies have better performance, as their decision variables in the optimization problem are only departure time and flight speed, respectively. The rerouting strategy is the slowest one and cannot even get converged within 500 iterations. The reason is that this strategy has three decision variables (x, y, z), and each of the variables will have a bigger value range when airspace size becomes larger. That increases the algorithm searching space and the optimization process needs more generation to find desirable solutions. The convergence speed of the ADM method is the second fast one, as most of the strategies employs are the scheduling and speed adjustment (see Fig. 6.7).

Table 6.2: Comparison results of different conflict resolution strategies.

Strategies	Total cost index	Number of conflicts	Number of flight delays
Scheduling	56208.46	9	58
Speed adjustment	38764.99	11	**0**
Rerouting	74348.46	13	22
ADM method	**32697.21**	**0**	6

As we can see from Fig. 6.7, the number of Speed strategy used notably increases while the number of the other two strategies decreases. After optimization, the number of speed adjustments used in the ADM method is 88, while the scheduling and rerouting strategies are 9 and 3, respectively. This indicates that the most suitable strategy for most conflicts is the Speed strategy. Because changing flight speed is effective to solve conflicts and it brings the least impact on a global flight plan, as it does not change the departure time and pre-planned 3D routes. However, there are still some flight conflicts the Speed strategy is unable to handle. For instance, the conflict cluster situation (as shown in Fig. 6.4(b), at the position of (x: 2000-5000, y: 3000)). In that case, the scheduling strategy is more suitable to solve the conflict clusters by adjusting the departure time of the pertaining UAVs. In addition, the Speed strategy is also invalid in another type of conflict, which is the head-to-head conflict (as shown in Fig. 6.2(c), the conflict of UAV c and UAV d). The most effective strategy for that case is rerouting, which changes the flight altitude or diverts 2D coordinates to avoid conflicts (as shown in Fig. 6.5(d)).

In this Chapter, there are multiple parameters used in the algorithm and the mathematical model. They can be classified as algorithm-related parameters and mathematical model-related parameters. The algorithm-related parameters are the basic ones that influence the solution quality and computational efficiency of the algorithm for all application cases. The sensitivity analysis for these parameters has been well conducted in the base SFS algorithm [194], and will not be covered in this Chapter. While for the model-related parameters, a sensitivity analysis is needed to investigate the influences of parameters on the optimization results of the multiple 4D flight routes. The model-related parameters include the constraint thresholds and the weightage factors, which are explained and presented in Table 6.1.

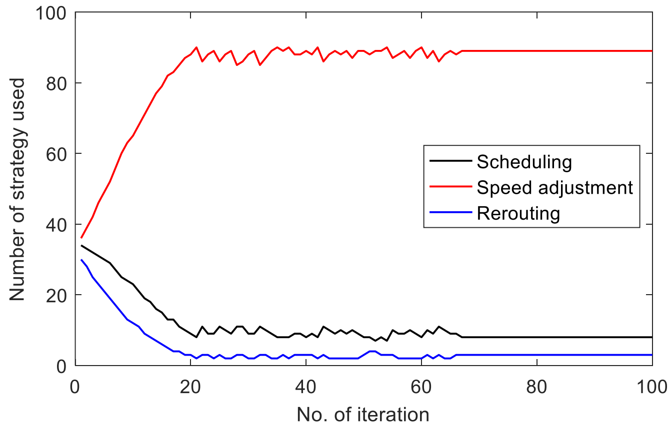


Fig. 6.7: Number of strategies used in the ADM method.

6.5.3 Sensitivity Analysis for Parameters Used in Proposed Model

6.5.3.1 Parameters in the thresholds of constraints

A. Threshold of separation minima

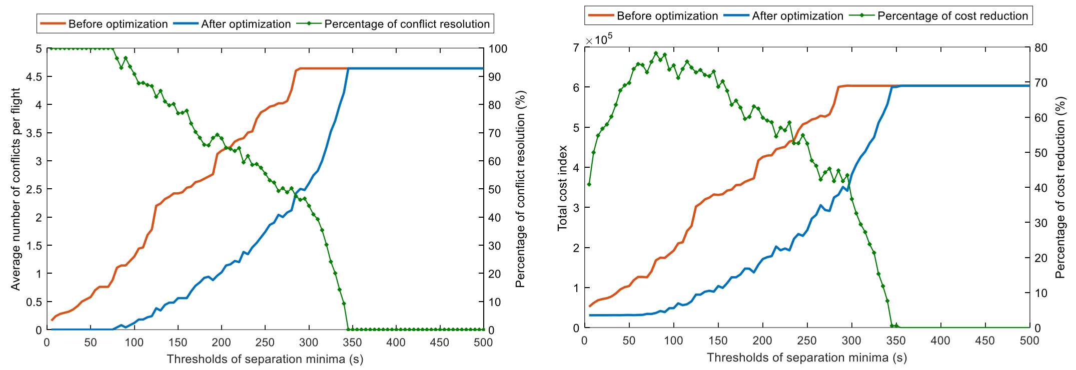
The thresholds of constraints include separation minima, battery duration, and departure delay. The determination of the threshold for separation minima is a compromise between the safety and efficiency of air traffic. A larger separation between aircraft makes a safer flight operation. However, it reduces the efficiency of air traffic flow and airspace utilization, as a larger separation requires a bigger safety buffer and conflict protection zone for each flight, which occupies more airspace. For the threshold of battery duration, it is determined by the battery capacity of the UAV. A better battery duration enables UAVs to execute a flight path with a longer distance. Lastly, the threshold of departure delay is determined by the user's acceptance of the time of delay. A larger threshold of departure delay can increase the punctuality rate of flights, as it increases the tolerability for the delay.

In the study, the separation minima are taken from the range of [5, 500] seconds with an interval of five seconds. The thresholds of battery duration and departure delay are taken from the range of [2, 100] minutes with an interval of two minutes. Sensitivity analysis is conducted for all three types of thresholds.

The threshold of separation minima is directly correlated with the number of flight conflicts. As discussed above, increasing the threshold for separation minima will increase the number of conflicts given the same density of air traffic. While the threshold does not correlate with flight delays, as changing the separation minima will not affect the departure time and arrival time. The sensitivity analysis results concerning an average number of conflicts and the total cost index for separation minima are presented in Fig. 6.8.

Fig. 6.8(a) shows the average number of conflicts with the changes in the separation minima thresholds before and after optimization. The percentage of conflict resolution after optimization is also presented. Before optimization, with the increase of the separation threshold, the average number of conflicts significantly increases. That is because a larger threshold causes more violations of separation minima, which produces more flight conflicts. While the average number of conflicts ceases to increase once the separation minima exceed a certain threshold. The reason is that the threshold of separation minima is too large, and the time difference between flights that will pass the same crossing waypoints is no longer smaller than the separation minima, as defined in Eq. (6.1). In this study, the threshold of separation minima is 290 seconds, and the average number of conflicts under that threshold is 4.64. After optimization, the average number of conflicts is significantly reduced. When the minimum separation is below a certain threshold, all conflicts can be solved (100% resolution) by using the adaptive strategies developed in this Chapter. In this study, the threshold for zero-conflict (no conflict) is 75 seconds with the traffic density being 100 flights/hour in the environment with a size of 36 km^2. Passing that threshold, the conflicts cannot be completely solved. With the further increase of the threshold, the average number of conflicts increases exponentially while the percentage of conflict resolution drastically decreases to zero when exceeding the full-conflict (no more conflicts) threshold of 345 seconds in this case.

Fig. 6.8(b) shows the total cost index with the changes of the separation minima thresholds before and after optimization. The results of the total cost index follow the same trends as the average number of conflicts, as the cost of flight conflict is the only contributor to the total cost index in the analysis of separation minima thresholds. While the percentage of the cost reduction shows a different trend. It increases significantly before exceeding the zero-conflict threshold (75 seconds) and then decreases to aero when exceeding a full-conflict threshold (345 seconds). The percentage of cost reduction is defined as $(C_{\mathtt{beforeOpt}} - C_{\mathtt{afterOpt}})/C_{\mathtt{beforeOpt}}$. Note that $C_{\mathtt{beforeOpt}}$ and $C_{\mathtt{afterOpt}}$ are total cost before and after optimization, respectively. Before exceeding the zero-conflict threshold, all conflicts can be solved and the $C_{\mathtt{afterOpt}}$ does not increase. While the $C_{\mathtt{beforeOpt}}$ continues to rise with the increase of the separation threshold. That makes the percentage of the cost reduction an increasing trend. After exceeding the zero-conflict threshold, both the $C_{\mathtt{beforeOpt}}$ and $C_{\mathtt{afterOpt}}$ begin to rise as the number of conflicts increases, which results in a decreasing trend for the percentage.



(a) Average number of conflicts per flight (b) Total cost index

Fig. 6.8: Sensitivity analysis for the thresholds of separation minima.

In summary, the conflict resolution and optimization results are highly sensitive to the threshold of separation minima. Exceeding a certain threshold of separation minima, the algorithm will not be able to solve all conflicts. With the further increase of the threshold, the average number of conflicts will begin to rise. The total cost index follows the same trend as the average number of conflicts shown.

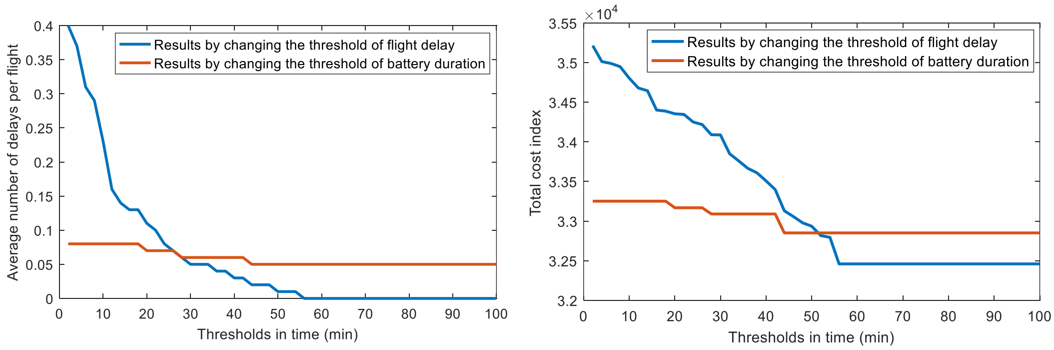
B. Thresholds of flight delay and battery duration

The sensitivity analysis for the thresholds of flight delay and battery duration is conducted. The threshold of flight delay is used to evaluate the number of delays, and the threshold of battery duration is used as a constraint for flight distance. Both factors do not correlate with the number of flight conflicts. These two thresholds will be evaluated by the average number of delays and the total cost index. The obtained results are presented in Fig. 6.9.

As defined in Eq. (6.4), a departure delay is determined by the difference between the actual departure time and the pre-planned one. In the optimization process, the actual departure time of some flights could be significantly changed to resolve conflicts. That will yield flight delays if the time difference exceeds the threshold of the departure delay.

As shown in Fig. 6.9(a), the average number of delays is sensitive to the threshold of the flight delay. With the increase of the threshold, the number of flight delays exponentially decreases all the way to zero when the threshold exceeds 56 minutes in this case study. However, 88% of the flight delays are within the threshold of 20 minutes (the average number of delays is 0.12), as the overall delay is to be minimized in the optimization process defined in Eq. (6.4). The total cost index shows a linearly decreasing trend (Fig. 6.9(b)) with the increase of flight delay threshold, as the number of flight delays is a contributor to the calculation of the total cost index. When the threshold exceeds 56 minutes, the delays are reduced to zero and the total cost index gets converged.

The analysis for the threshold of battery duration shows that the optimization results are not sensitive to that threshold. Both the average number of delays and the total cost index have no significant changes with the increase of the battery duration threshold, as shown in Fig. 6.9(a) and Fig. 6.9(b). That is because the battery duration is only considered as a constraint for airborne time, as defined in Eq. (6.8). The airborne time is correlated with the flight distance and the flight speed of a route. To avoid conflicts, a long flight distance or a very slow flight speed could be assigned, which will increase the airborne time of UAVs for a specific route. In our proposed model, a strategy of generating a long flight route will be discarded in the optimization process, as a longer distance increases both the operational risk (Eq. (6.3)) and airborne time (Eq. (6.5)) that are minimized in the objective function. The slow speed (or even hovering) can also be avoided by the constraint given for the range of the flight speed (Eq. (6.11)). That all makes the results not sensitive to the battery duration threshold.



(a) Average number of delays per flight (b) Total cost index

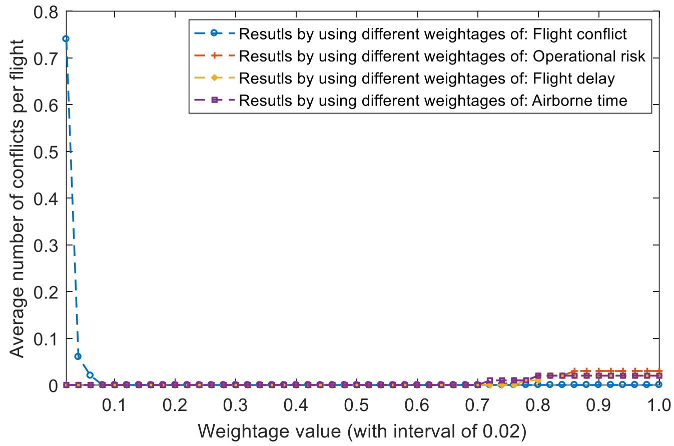
Fig. 6.9: Sensitivity analysis for the thresholds of flight delay and battery duration.

6.5.3.2 Weightage factors in the objective and penalty functions

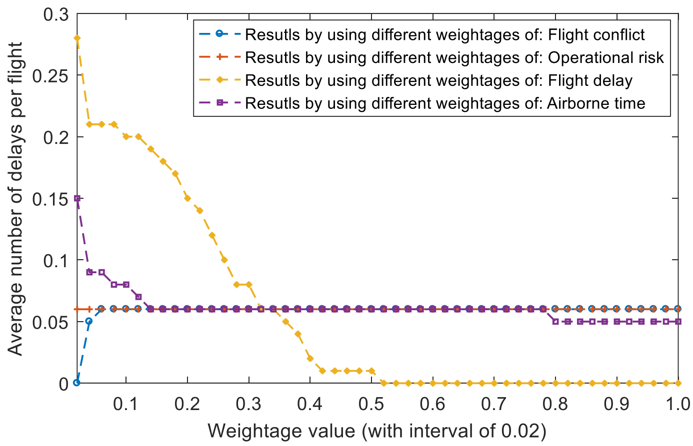
The weightage factors used in the objective function and penalty function may also influence the optimization results. These weightage factors are the weightage of flight conflict ω_{fc}, weightage of operational risk ω_{risk}, weightage of flight delay $\omega_{\mathrm{t\_delay}}$, and weightage of airborne time $\omega_{\mathrm{t\_air}}$. In practical applications, these factors can be determined by users' preferences or the significance of the factors. In this analysis, all factors are taken from the range of [0, 1] with an interval of 0.02 to demonstrate the sensitivity of optimization results with the changes of these factors. As the factors of ω_{risk}, $\omega_{\mathrm{t\_delay}}$, and $\omega_{\mathrm{t\_air}}$ are inclusive (Eq. (6.2)) in the computation of the total cost index, they will be evaluated by the average number of conflicts and the average number of delays. The constraint thresholds are used as the ones given in Table 6.1, and they are fixed in the analysis. Only the weightage factors will be analyzed in this subsection. The obtained results are presented in Fig. 6.10.

The average number of conflicts is highly sensitive to the weightage of flight conflict, as shown in Fig. 6.10(a). There is a giant drop in the average number of conflicts when the weightage of flight conflict comes into play. When the weightage ω_{fc}=0, the number of conflicts is not reduced, as there is no penalty for flight conflicts in the optimization process. While when ω_{fc}=0.02, the average number of conflicts is dramatically reduced from 0.74 to 0.05, as the penalty is introduced to discard solutions with flight conflicts. Though the weightage is small, it is the times of the total cost index, which makes the penalty effective. The more conflicts a solution has, the higher probability that the solution will be discarded in the next iteration. On the other hand, the average number of conflicts is not sensitive for the rest of the weightage factors, as they do not contribute to the penalty that is used to penalize the solutions with conflicts. However, these weightage factors have a knock-on effect for conflict resolution when they become large. For instance, when the weightage of the operational risk exceeds 0.86, the average number of conflicts increases from 0 to 0.03, which is equivalent to three conflicts out of 100 flights. The reason is that too high an operational cost will deny the rerouting strategy, as it will increase the cost when changing the route from the original cost-optimal one.

Come to the average number of flight delays, it is highly sensitive to the weightage of the flight delay, as presented in Fig. 6.10(b). With the increase of this weightage, the average number of delays has significantly decreased. That is because a larger weightage of delay will make the delay cost higher. Solutions that use the scheduling strategy to delay the departure time will be reduced and that reduces the average number of delays. While the analysis results are less sensitive for the weightage of airborne time and flight conflict, compared with the weightage of delay. With the increase of the two weightage factors, the minimization of airborne time will dominate the objective function. Solutions with shorter airborne time will be selected, which increases the punctuality of ETA, and subsequently reduces the number of delays. In contrast, the average number of delays increases with the increase of the flight conflict weightage. The scheduling strategy would be used to solve a certain type of flight conflict when the weightage of conflict comes into play. For the last weightage factor of operational risk, the number of delays is not sensitive to it. As the operational risk is only correlated with the waypoints of the routes and does not influence the delays.



(a) Average number of conflicts per flight



(b) Average number of delays per flight

Fig. 6.10: Sensitivity analysis for the weightage factors.

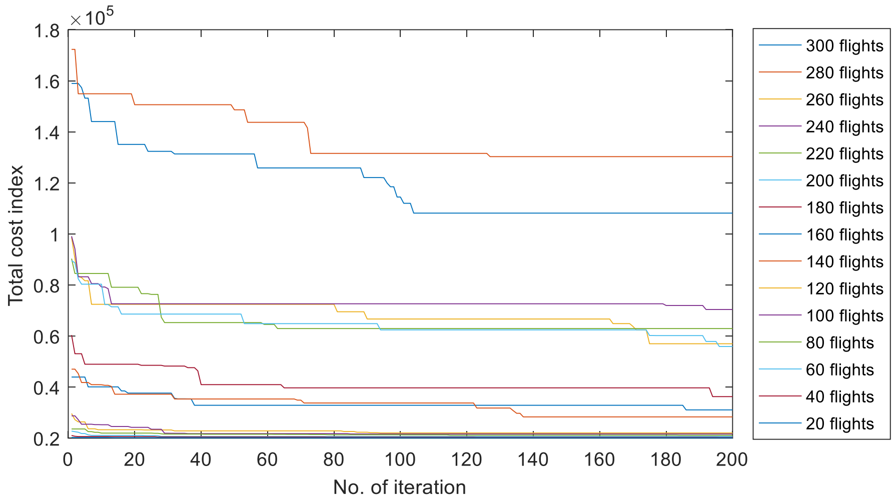
6.5.4 Algorithm Testing in Different Air Traffic Densities

Simulations are conducted to investigate the optimization performance of the proposed algorithm in scenarios with different air traffic densities. The highest density in this simulation is 300 drones per hour in an environment with a size of 36 km^2. That is equivalent to 6,000 drone operations per hour (144,000 per day) in Singapore. Simulations are performed in the same environment with an increasing number of flights from 20 to 300 with an interval of 20. The ISFS algorithm is used to optimize the 4D routes, and the obtained results are presented in Fig. 6.11 and Fig. 6.12.

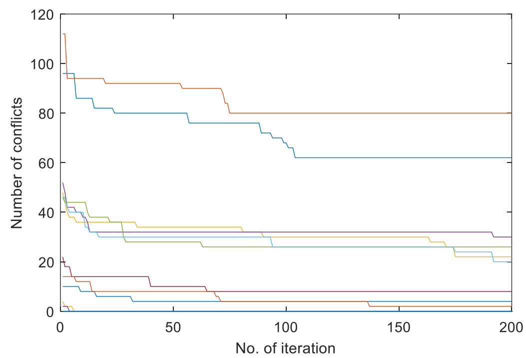
In general, with the increase of flight density, the total cost index (Fig. 6.11(a)), number of flight conflicts (Fig. 6.11(b)), and number of flight delays (Fig. 6.11(c)) increase. It is also notable that within a range of flight numbers, the performance indicators have no significant changes. However, from one range to another, there is a great jump in terms of performance change. For instance, as we can see from the results in Fig. 6.11, there are four ranges of flight number, which are [20, 120], [140, 180], [200, 260], and [280, 300] with an interval of 20. In the first range, all indicators are convergent. With the increase in traffic density, the algorithm is still effective, but it needs more iterations to get converged.

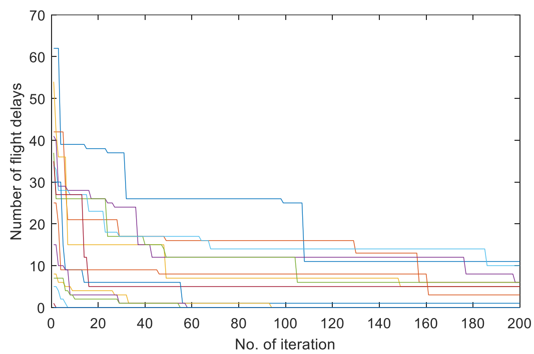
In addition, the case of 300 flights has an even better performance than that of the 280 flights case, which reveals another finding: there has a clear threshold of airspace capacity, which can be evaluated by the operational performance. For instance, the increase in flight numbers within a certain group (e.g. [140, 180]) will not significantly influence traffic performance in terms of operational cost and flight conflicts. That is because, within a certain range, the proposed ADM algorithm can always find suitable strategies to handle flight conflicts and optimize the 4D routes. However, once the flight density exceeds a certain level of capacity threshold, the newly added flights will cause a significant impact on existing flight plans. After that, a new balance is established until the number of flights exceeds the next threshold. This finding can facilitate the determination of the airspace capacity threshold.

Note that Fig. 6.12 shows the trends of the average cost index, rate of flight conflicts, and rate of flight delays. All the indicators increase exponentially with the increase in flight density. In the first range [20, 120], the average cost index and the number of flight delay slightly increases while there is no flight conflict. In this range, the traffic density is low, and the ADM method provides good solutions for all conflicts with only a few flights exceeding the arrival delay threshold. That is because the scheduling strategy is used for a portion of conflicts, which may postpone some flights causing delays.



(a) Total cost index

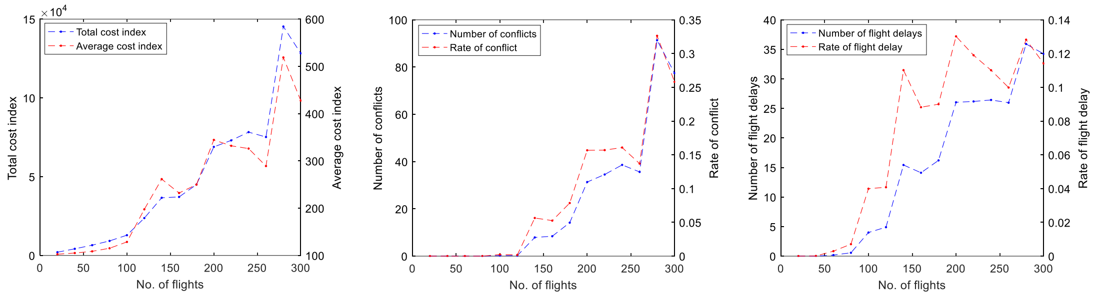




(b) Number of flight conflicts (c) Number of flight delays

Fig. 6.11: Optimization performances with different traffic densities. The traffic density is defined as the number of flights per hour in an environment with a size of 36km^2.

Come to the second range [140, 180], there is a rise in the indicators with an average cost index increases to 250 (Fig. 6.12(a)), the rate of conflict increases to 0.05 shown in Fig. 6.12(b), and rate of flight delays to 0.1 shown in Fig. 6.12(c). A steeper increase is from the second range to the third range, with the average cost index increasing from 200 to 300 and the conflict rate from 0.05 to 0.15. The most significant jump is from the third range [200, 260] to the fourth range [280, 300], with the average cost index and rate of conflict doubled.



(a) Cost index (b) Flight conflicts (c) Flight delays

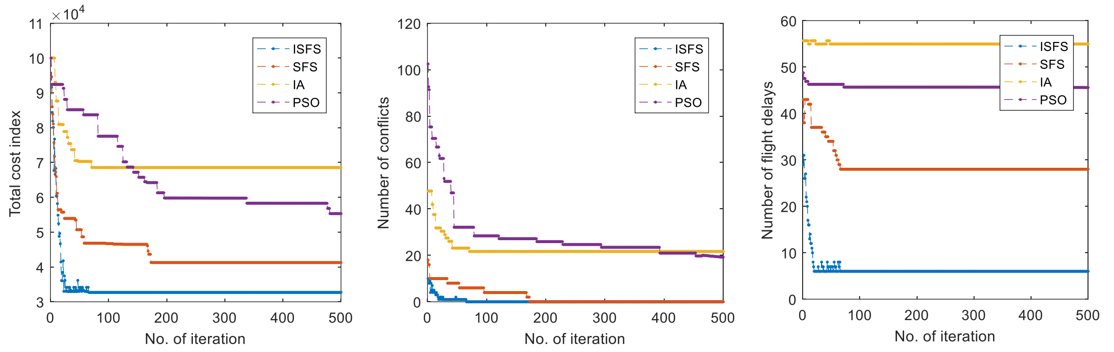
Fig. 6.12: The average performance of indicators decreases with the increase of traffic density. The traffic density is defined as the number of flights per hour in an environment with a size of 36 km^2.

6.5.5 Comparison of Algorithms: ISFS, SFS, IA, and PSO

To verify the performance of the proposed ISFS algorithm, we compare ISFS with other three well-known meta-heuristic search algorithms, which are the original stochastic fractal search (SFS), immune algorithm (IA), and particle swarm optimization (PSO) algorithm. The population size is taken as 50 for all algorithms and the number of iterations is taken as 500. The simulation environment is the same as the one we developed in Section 6.5.1. The number of flights is taken as 100 and the Dijkstra algorithm is used to generate the initial 4D routes as input for the optimization.

Fig. 6.13 shows the optimization results by using the four algorithms. Our proposed ISFS algorithm performs the best in all indicators and has the fastest convergence rate. It has the lowest cost index (Fig. 6.13(a)) and has no flight conflict in optimized 4D routes (Fig. 6.13(b)), while the number of flight delays is also the smallest (Fig. 6.13(c)). Followed by the SFS method, it has the second lowest cost index and has no flight conflict after optimization. The PSO algorithm does not converge whereas its performance is good in minimizing the cost index and the number of conflicts. As for the IA method, although it has a fast convergence rate, the cost index of this method is two times more than that of the ISFS algorithm, and it has the largest number of flight delays among the four algorithms.

To test the fairness of the algorithm comparison results and validate the reliability of the proposed ISFS method, we conducted 50 times of independent simulations with the same initial 4D routes in the same environment. The obtained statistical results are shown in Table 6.3.



(a) Total cost index (b) Number of flight conflicts (c) Number of flight delays

Fig. 6.13: Algorithm comparison results among ISFS, SFS, IA, and PSO.

On average, the ISFS is the best among the four algorithms in terms of solution quality and convergence time. It has the lowest average cost index (32697.21) with the smallest standard deviation (1594.87). The convergence time of the ISFS is also the lowest (216.59 s). The optimized 4D routes using ISFS have no conflict in all 50 simulations and the average number of flight delays is the smallest (6 delays). The results of the SFS method have an increase of 16.18% in cost index, 0.87% in standard deviation, and one flight conflict, compared with ISFS. The results produced by the PSO algorithm have a dramatic increase in cost index, flight conflicts, and delays with the largest standard deviation (6857.87). The performance of the IA method is down to the bottom of the list, with the average cost index more than doubled compared with ISFS. The number of flight conflicts and delays is the highest with 22 and 55, respectively.

Table 6.3: Statistical results (on average) of the 50 independent simulations.

Algorithms	Cost index	Standard deviation of cost index	Number of conflicts	Number of flight delays	Convergence time (s)
ISFS	**32697.21**	**1594.87**	**0**	**6**	**216.59**
SFS	37988.46	1608.74	1	28	601.92
IA	68557.40	4103.85	22	55	638.92
PSO	55330.20	6857.87	19	46	803.55

6.5 Summary

This Chapter studies the flight conflict resolution problem for risk-based UAV 4D routes optimization in urban environments. A double-layer decision-making framework is proposed to optimize multiple 4D routes, and an improved stochastic fractal search (ISFS) algorithm is developed to solve the problem. Simulations and comparison studies are performed to demonstrate the performance of the proposed model and algorithm. The main findings of this Chapter are concluded as follows.

First, the adaptive decision-making method can solve flight conflicts and has better performance than individual strategy (e.g., scheduling or rerouting), as it selects the best suitable resolution strategies for different types of conflicts. Among the three individual strategies, the speed adjustment has the best performance in solving conflicts with the least disturbance caused to global planned routes. That is because the speed adjustment does not change the departure time and the initially-planned routes.

Second, to optimize the risk-based 4D routes with assigned conflict resolution strategies, an MINLP model is proposed, and an improved stochastic fractal search algorithm is developed. By introducing the exploitation-exploration balancing scheme, the improved ISFS algorithm performs the best in reducing operational cost, flight conflicts and delays compared with the original SFS, IA, and PSO algorithms. Its performance is enabled by the improved fitness-distance balancing scheme that is used in the update processes to avoid exploring points that are too close to the current best point, which improves the searching diversity of ISFS.

Third, the sensitivity analysis on the number of flights indicates that the proposed ISFS algorithm is effective in dealing with conflicts in different traffic density scenarios. There is also a clear trend that the performance indicators of 4D routes have no significant changes within the range of flight numbers. However, from one range to another, there is a great jump in terms of performance decreasing. That could be a useful finding to support the determination of airspace capacity using performance-based thresholds.

Chapter 7

7 Robust Unmanned Air Traffic Flow Management (UTFM) under Uncertainties

In Chapter 6, we study the multi-UAV 4D trajectory optimization problem with strategic deconfliction to produce conflict-free 4D trajectories. However, the strategic conflict-free trajectories could be disrupted due to uncertainties of flight delays and deviations of waypoint occupancy times. To cope with these uncertainties in conflict resolution decisions, this Chapter proposes a chance-constrained urban air mobility (UAM) traffic flow management (UTFM) optimization model to generate robust conflict-free trajectories and reduce flight delays. The model includes a probabilistic constraint of separation requirement to cope with trajectory time deviations caused by environmental disturbances, such as wind. To solve the model, we convert the probabilistic constraint into a deterministic constraint with a risk-bounded separation guarantee. Numerical studies demonstrate the proposed model's effectiveness, robustness, and scalability. The proposed chance-constrained UTFM model forms an additional layer of safety assurance in the pre-tactical phase, which enables fast flight disruption recovery in the UTFM system under delay and trajectory uncertainties.

7.1 Introduction

To manage the complex and high-density UAM traffic flow safely and efficiently, several concepts of operations have been proposed. Free route airspace (FRA) [229] provides full flexibility and could accommodate high-density UAM operations. However, it only provides one safety assurance layer (tactical detection and avoidance), which may make the operations unacceptable due to the stringent target level of safety for the safety-critical UAM system [230]. The corridor-based operation, as a conservative concept [135], provides strong safety assurance by limiting eVTOLs operating only within the corridor, which makes it difficult to satisfy the entrance and exit demands of UAM operations that have flexible origins and destinations. Trajectory-based operation (TBO) combines both advantages of safety assurance and flight flexibility, which is a fundamental concept to managing high-density air traffic in the Next Generation (NextGen) Air Transportation System [231]. As a core tenant of TBO, the aircraft's four-dimensional trajectory (4DT) plays a key role in enhancing the safety, efficiency, capacity, and resiliency of air traffic flows. By generating conflict-free 4DTs using strategic deconfliction [232], potential conflicts and congestion can be avoided before flight. However, the effectiveness of strategic planning could be significantly degraded due to uncertainties such as departure delays and environmental disturbances [230].

Departure delays, on the one hand, will directly cause arrival delays given the same flight speed, which starts a wave of delay propagations for the following flight legs [105]. On the other hand, departure delays cause a complete change in the estimated time of arrival (ETA) for all crossing waypoints (CWPs) in the delayed 4DT [233]. The changed ETA of the CWPs may yield new conflicts, as these CWPs were planned to be used and occupied by other 4DTs at that time. In real-world operations, the waypoint occupation time (WOT) of these CWPs can be probabilistically distributed due to environmental disturbances, especially the wind [234]. The probabilistic WOT fails the traditional conflict detection approaches that rely on a determined separation minima. Consequently, the traditional 4DT optimization methods with such a deterministic separation constraint become no longer applicable under uncertainties.

The above-discussed uncertainties could disrupt the pre-planned conflict-free 4DTs, yielding new conflicts caused by delays and miss-detected conflicts due to probabilistic waypoint occupation time. That breaks the strategic safety assurance layer and cascades too many conflicts that may not be solved by the tactical layer (on-board DAA) alone at an acceptable risk level (e.g., 10^{-6} fatalities/flight hour) [27]. It is, therefore, crucial to reconstruct the pre-flight safety assurance layer by recovering the disrupted 4DTs with resilient and conflict-free 4D trajectories in response to observed disruptions and future uncertainties.

7.1.1 Studies on Air Traffic Flow Management under Uncertainties

As a safety-critical system, the air traffic system has three safety assurance layers in the strategic, pre-tactical, and tactical flight phases [235]. The scope of this Chapter is on the pre-tactical phase with the aim of optimizing the disrupted conflict-free 4DTs that are generated in the strategic phase. The disrupted 4DTs are optimized in two procedures: (i) robust 4D trajectory optimization (to improve its resilience to future trajectory uncertainties), which is considered a proactive approach; (ii) disruption recovery (to optimize the observed disruptions), considered as a reactive approach. The focus of the literature review is on these two aspects.

Aircraft 4D trajectory optimization problems have been extensively studied thanks to their advantages of improving the predictability of the traffic flow [232]. Among these, most of the work is done in the strategic phase to generate conflict-free 4DTs [142], which converts the optimization problem from four dimensions (latitude, longitude, altitude, and ETA) into one dimension (ETA of passing waypoints). This advantage makes the fast 4DT disruption recovery possible. However, the strategic 4DTs have their weaknesses in that the effectiveness could be significantly degraded due to uncertainties [230], resulting in disrupted 4DTs with conflicts and delays.

To cope with the aforementioned uncertainties, researchers have investigated the air traffic flow optimization problem [236] under various uncertainties. Capacity uncertainty is one of the main types, which could be impacted by weather conditions that are hard to be predicted [117], [233], [237]. To cope with the capacity uncertainty of airspace sectors, one study [237] formulated a chance-constrained optimization model with probabilistic sector capacities. An approximation-based algorithm was proposed, which allows efficient computation for large-scale chance-constrained optimization problems. A follow-up work [238] considered both departure and En route uncertainties to ensure no capacity violations while maintaining the in-trail separation under trajectory uncertainties. The trajectory uncertainties result in deviations at spatial and temporal dimensions [117]. The spatial deviations could lead to safety risks in which flight trajectories exceed the airway boundary, while time deviations (e.g., ETA deviation) may yield congestion and misdetection of conflicts [233].

To address separation violation risks under trajectory uncertainties, studies have been done on the development of robust conflict detection and resolution (CD&R) models under weather uncertainties with the objectives of minimizing delays, conflicts, and deviations from nominal trajectories [210], [239], [240]. In these studies, the weather uncertainties are described as random variables that follow a certain distribution. To obtain a more accurate uncertainty distribution, recent works [241], [242] employed data-driven approaches to predict flight trajectories in spatial-temporal dimensions. The proposed data-driven models demonstrated better prediction accuracy by utilizing the real-time observed weather information in the air traffic control system. However, air traffic disruptions are inevitable in uncertain environments and need to be mitigated to reduce the cost of flight conflicts, delays, and cancellations. In one study [243], dynamic disruption management was formulated as a dynamic stochastic integer programming model to minimize the network-wide recovery cost. Another study integrated aircraft and passenger recovery and formulated it as a mixed integer linear programming model to minimize the overall cost from both the airline and passenger sides [244]. The well-developed strategic ATFM and pre-tactical disruption management under uncertainties form two safety assurance layers, which significantly improve safety by reducing the number of interventions at the tactical level by air traffic controllers [233].

Studies done in ATFM under uncertainties inspire the solving of UTFM challenges posed by unique features of high-density traffic flow, on-demand missions, and time-sensitive decision making of the emerging UAM traffic flow [216], [245]. To accommodate the UAM traffic flow, an AirMatrix concept has been proposed in which the airspace is divided into standard 3D blocks [55], [142], [167]. These standard blocks allow digital information (e.g., weather, signal conditions, etc.) to be used for the optimization of trajectory-based operations [170]. Under the TBO concept, 4D trajectory optimization models were developed with strategic deconflicts to generate conflict-free 4DTs through strategies of rerouting, flight level change, and speed adjustment [110], [141], [246]. To cope with weather uncertainties, robust models for drone path planning [234] and 4D trajectory optimization [114] were developed by considering probabilistic time deviations [247]. The above-discussed literature solves the 4DT optimization problem in the strategic phase, while the effectiveness of the strategic planning could be disrupted due to uncertainties (e.g., delays) that could only be known after the nominal departure time [230]. To minimize the disruption cost, ATFM fields have investigated the disruption management problem, however, disruption recovery studies are still lacking in the UTFM domain. It becomes even more challenging for the on-demand UTFM which requires fast decision-making within minutes [248], compared with the ATFM during the day of operations [243].

Tactical decision-making handles flight disruptions in real-time manners, which forms another safety assurance layer for safe UAM operations. There is a series of pioneer works [216], [249] that have been done on UAM tactical CD&R with learning-based approaches. The CD&R problems were formulated as a Markov Decision Making process, while Monte Carlo tree search and reinforcement learning approaches were employed to solve the developed problems in real time. To cope with the position uncertainty in real-time CD&R, risk-bounded separation models were introduced to generate more robust trajectories at a given risk level [249], [250]. The performance of deep reinforcement learning approaches has also been demonstrated in trajectory optimization for heterogeneous autonomous vehicles. Where multilayer network structure was used to capture the system's complexity, while transfer learning techniques were developed and applied to enable planning performance for various types of vehicles [251], [252]. Their simulation and experiment results demonstrated the good performance of the models in applications of optimal parking maneuver and overtaking maneuver problems [253].

Digging further, a series of research works have been done to generate optimal trajectories for unmanned vehicles in real-time perspectives considering motion control uncertainties [254]–[257]. In these works, the probabilistic control actuation constraint and collision avoidance constraint were considered for fixed-wing UAV safe operations. The concerning problems were constructed as chance-constrained trajectory optimization models, and convex approximation techniques were introduced to convert the models into deterministic ones, which enables fast generation of safe UAV trajectories [254], [255]. To further handle multi-objective trajectory optimization problems, follow-up works developed hybrid swarm intelligent algorithms to solve the converted unconstrained models. The main contributions in these works were the introduction of novel selection methods in exploration and exploitation processes [256], [257].

Besides, authors [258] also modeled the capacity uncertainty of vertiport using mixed integer programming techniques to minimize congestion in the landing phase [258]. A follow-up work [259] further explored the uncertainty of Geo-Fencing boundaries, and a chance-constrained optimization model was developed to minimize the probability of collision between UAVs and Geo-Fences (e.g., buildings). Apart from the safety and efficiency objectives, researchers considered that fairness is equally important and should be improved when making decisions [250], [260]. Their simulation results demonstrated that fairness can be significantly improved with only little cost to the system efficiency if the fairness metric could be properly modeled.

Prior literature has investigated the air traffic flow management problem for both manned and unmanned flights at strategic and tactical levels. However, models are lacking for the disruption recovery of UAM traffic flows that are more vulnerable under uncertainties of individual flight missions and challenging weather conditions in low-altitude urban airspace [234]. Although robust traffic flow optimization has been well recognized as an important aspect of ATM, it has only recently been investigated for UAM operations with uncertainty modeling of vertiport capacity [258] and position deviations [250]. The modeling for time uncertainty of eVTOL's trajectory has not been explored, especially in complex urban environments where weather and navigation conditions could lead to significant trajectory ETA deviations [19], [144], [261]. The time-sensitive UAM flight missions require an even faster decision-making tool, posing a challenge for solution algorithms, and an even bigger challenge is posed by the fairness requirement in that the decision space is reduced as non-delayed flights will not be adjusted [260].

7.1.2 Research Scopes

To solve these unique challenges, this Chapter develops a chance-constrained UTFM optimization model with fast disruption recovery considering the ETA deviation that is derived from a proposed data-driven uncertainty distribution. This Chapter conducts three key research works for safe and resilient UAM traffic flow management.

(i) We propose a chance-constrained UTFM optimization model with fast disruption recovery to solve observed disrupted 4DTs (reactive approach) and improve their resilience in response to future uncertainties (proactive approach). Which forms another safety assurance layer for the UAM traffic system.

(ii) We develop a probabilistic separation requirement model to adaptively determine the risk-bounded separation for flights in different uncertainty levels. The model enables the connection between the decision maker's risk preferences and the choice of the trajectory uncertainty distribution.

(iii) We propose a hierarchical stochastic search (HSS) algorithm to solve the developed optimization model in polynomial time with a safety guarantee of resolving all conflicts, making it a practical solution algorithm for UAM traffic flow optimization and disruption recovery.

The rest of this Chapter is organized as follows: the problem statement and definition of parameters used in the UTFM model are presented in Section 7.2, followed by the formulation of the UTFM optimization problem under chance constraints in Section 7.3. The solution approaches are developed in Section 7.4, followed by the numerical analysis and discussions in Section 7.5. The summary is discussed in Section 7.6.

7.2 Problem of Decision Making under Uncertainties

In this section, we illustrate the conflict-free 4D trajectory and how it will be disrupted due to flight delays and probabilistic waypoint occupation time caused by trajectory deviations.

7.2.1 Disrupted Conflict-free 4DTs Due to Delays

Aircraft's four-dimensional trajectory is defined in dimensions of latitude, longitude, altitude, and time [231]. The potential conflicts among multiple 4DTs can be resolved using the deconfliction method [141], which generates conflict-free 4DTs before flight. However, the conflict-free 4DTs could be disrupted by flight delays, resulting in new conflicts. As illustrated in Fig. 7.1(a), a delayed flight f_2 arrives later (at $t_1{}'$) than the initially assigned time t_1. The delayed arrival causes a conflict with a flight f_1 at a crossing waypoint w_{cp}. The f_2 may travel multiple crossing waypoints and cause more conflicts with other flights due to the delayed arrivals, as shown in Fig. 7.1(b). The initial conflict-free 4DT of flight f_2 has no conflicts with other 4DTs at crossing waypoints. While the delayed arrival creates new conflicts as it changes the ETAs at the crossing waypoints. An example is illustrated in Fig. 7.1(c). The conflict-free 4DT of flight f_2 delayed from t_1 to t_3, resulting in the subsequent changes of ETAs of all its waypoints. The changed ETAs at crossing waypoints $(w_{\text{cp\_2}}, w_{\text{cp\_4}}, \ldots, w_{\text{cp\_}n})$ clash with the waypoint occupation time that are used by other 4DTs, creating conflicts.

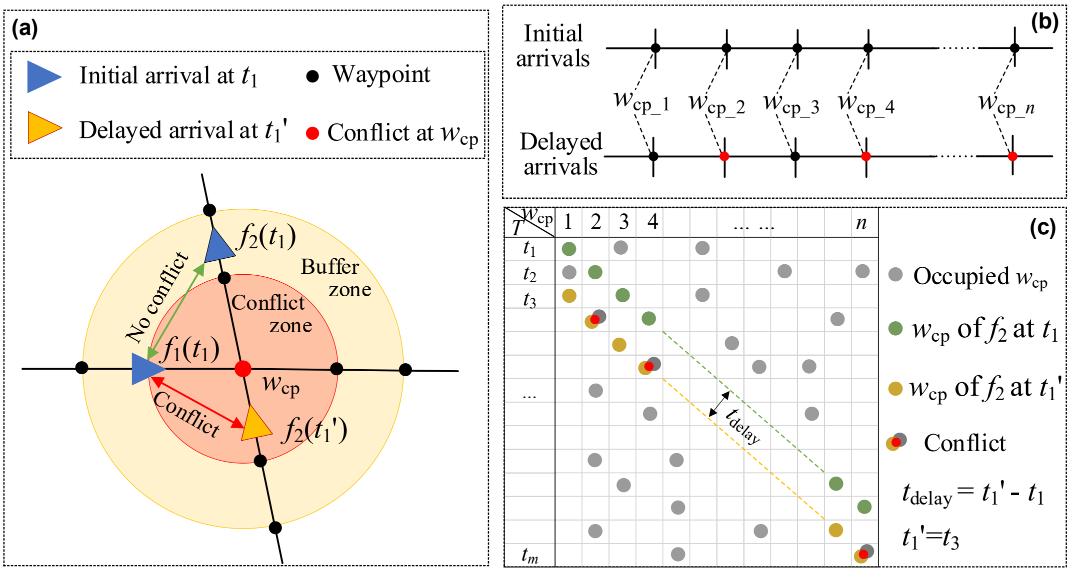


Fig. 7.1: Disruption of conflict-free 4D trajectories due to departure delays. (a) A conflict caused by a delayed flight f_2 at a crossing waypoint. (b) Flight conflicts caused by delays. The initial conflict-free 4D trajectory has no conflicts at all crossing waypoints, while the delayed 4DT causes new conflicts. (c) An example of ETAs of a delayed flight f_2 clash with waypoint occupation time at several crossing waypoints.

7.2.2 Probabilistic Uncertainty of Waypoint Occupation Time (WOT)

The ETAs of delayed flights may clash with the WOT where the waypoints are used by other flights. The WOT is equivalent to a determined separation requirement if there are no trajectory deviations, as the WOT is the minimum required time for a flight passing the waypoint after the preceding flight left. In real-world operations, trajectory deviations are inevitable due to environmental disturbances [20] that make the WOT a probabilistic uncertainty. As illustrated in Fig. 7.2, the time of a flight arriving at a crossing waypoint before WOT is named ETA1, while the one after WOT is named ETA2. No conflict is generated on ETA1 and ETA2 at the crossing waypoint under a deterministic WOT. However, under a probabilistic WOT, the ETA1’ clashes with the probabilistic WOT before entering the waypoint while ETA2' clashes after passing the waypoint, causing conflicts. As WOT is caused by trajectory deviations in the temporal dimension, the distribution of WOT can be derived from the historical trajectory deviation data. That will be used for the robust UTFM optimization.

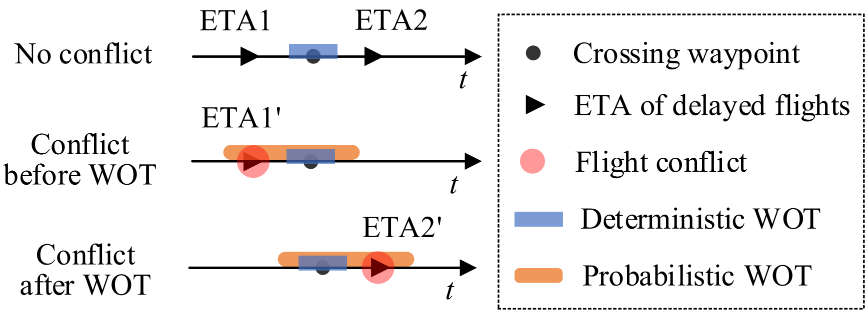


Fig. 7.2: Probabilistic uncertainty of the waypoint occupation time.

7.2.3 Expression of Parameters and Terms

The scope of this Chapter is on UTFM optimization with fast disruption recovery to solve conflicts and delays caused by uncertainties. Different from scheduled commercial flights, UAM operations are more flexible, and include both scheduled commuting services [186] and on-demand missions [262]. To optimize the delayed flights and cope with the on-demand operations within a short time, the pre-tactical phase for UAM operations is defined as [1, 10] minutes prior to the initial departure time. To ensure fairness, only delayed flights will be optimized by adjusting their speed and actual departure time. The non-delayed flights will operate as usual with initially planned 4D trajectories. By doing so, the second-order conflicts [239] will be avoided, and the robustness of the whole system can be improved. Under the pre-tactical phase for UAM operations, we define the parameters and terms used in this Chapter.

Definition 1 (Arrival Delay): Let f_k be the kth delayed flight. The initial arrival time and actual arrival time of f_k to the destination can be denoted as $t_{f_k}^{\text{arr}}$ and $t'^{\text{arr}}_{f_k}$. The arrival delay $t_{f_k}^{\text{arr\_dly}}$ of flight f_k is defined as the difference between $t_{f_k}^{\text{arr}}$ and $t'^{\text{arr}}_{f_k}$, denoted as

$$t_{f_k}^{\text{arr\_dly}} = t'^{\text{arr}}_{f_k} - t_{f_k}^{\text{arr}} \tag{7.1}$$

where $t_{f_k}^{\mathrm{arr\_dly}} \in R$. Note that the initial arrival time $t_{f_k}^{\mathrm{arr}}$ is a constant variable obtained in the pre-planned phase. While the actual arrival time $t'^{\mathrm{arr}}_{f_k}$ is correlated with the actual departure time $t'^{\mathrm{dep}}_{f_k}$ and the actual airborne time $t_{f_k}^{\mathrm{AAT}}$, denoted as

$$t'^{\mathrm{arr}}_{f_k} = t'^{\mathrm{dep}}_{f_k} + t_{f_k}^{\mathrm{AAT}} \tag{7.2}$$

The definitions of the postponed delay due to $t'^{\mathrm{dep}}_{f_k}$ and the actual airborne time $t_{f_k}^{\mathrm{AAT}}$ are given as follows.

Definition 1.1 (Postponed delay): Let $t_{f_k}^{\mathrm{dep}}$ be the initial departure time of f_k. The postponed delay $t_{f_k}^{\mathrm{ppn}}$ of f_k can be defined as

$$t_{f_k}^{\mathrm{ppn}} = t'^{\mathrm{dep}}_{f_k} - t_{f_k}^{\mathrm{dep}} \tag{7.3}$$

Note that the actual departure time of delayed flights should not be less than the initial departure time, denoted as $t'^{\mathrm{dep}}_{f_k} \geq t_{f_k}^{\mathrm{dep}}$, and $t_{f_k}^{\mathrm{ppn}} \in R_+$. The reason for postponing the already delayed departure flight is to resolve flight conflicts and reduce capacity-to-demand imbalance if other strategies failed to do so.

Definition 1.2 (Actual airborne time: AAT): Let $s_{ij\_f_k} \in S_{f_k}$ be a trajectory segment of the delayed flight f_k. The flight time of f_k on the segment $s_{ij\_f_k}$ can be denoted as $t_{s_{ij\_f_k}}$. The actual airborne time of kth delayed flight $t_{f_k}^{\mathrm{AAT}}$ is defined as

$$t_{f_k}^{\mathrm{AAT}} = \sum_{s_{ij\_f_k} \in S_{f_k}} t_{s_{ij\_f_k}} = \sum_{s_{ij\_f_k} \in S_{f_k}} \frac{l_{s_{ij\_f_k}}}{v_{s_{ij\_f_k}}} \tag{7.4}$$

where $l_{s_{ij\_f_k}}$ is the length of the segment s_{ij}, and $v_{s_{ij\_f_k}}$ is the flight speed of f_k on the segment $s_{ij\_f_k}$.

The correlations of the time-related terms mentioned above are illustrated in Fig. 7.3.

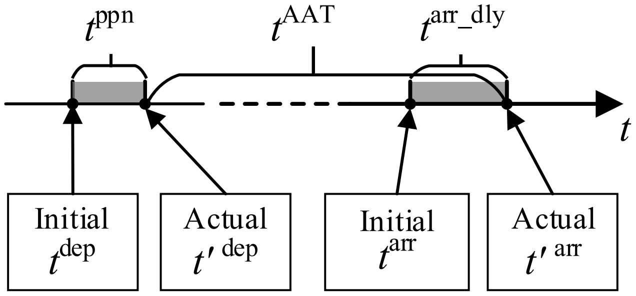


Fig. 7.3: Correlations among the time-related terms. Including the postponed time of delay t^{ppn}, actual airborne time t^{AAT}, and arrival delay $t^{\mathrm{arr\_dly}}$.

Definition 2 (*Flight conflict*): A flight conflict is detected if the difference between $t_{w_{\mathrm{cp}(i)}\_f_k}^{\mathrm{ETA}}$ (the ETA of a flight f_k on $w_{\mathrm{cp}(i)}$) and $t_{w_{\mathrm{cp}(i)}}^{\mathrm{WOT}}$ (the waypoint occupation time of $w_{\mathrm{cp}(i)}$) is smaller than the separation requirement t_{sep}, denoted as

$$\left| t_{w_{\mathrm{cp}(i)}\_f_k}^{\mathrm{ETA}} - t_{w_{\mathrm{cp}(i)}}^{\mathrm{WOT}} \right| < t_{\mathrm{sep}} \tag{7.5}$$

Note that $t_{w_{\mathrm{cp}(i)}}^{\mathrm{WOT}}$ is the ETA of a flight in a conflict pair, which comes first into the conflict buffer zone and occupies the waypoint at that period. Other flights should adjust and optimize their trajectories to avoid flight conflict.

With the defined flight conflict condition, the probabilistic uncertainty of the waypoint occupation time can be handled by guaranteeing that the probability of the flight conflict is controlled in an acceptable risk level $\alpha \in [0,1]$, which can be denoted as

$$\mathrm{P_r}(\mathrm{Conflict}) \leq \alpha \tag{7.6}$$

The Chapter aims to minimize flight arrival delays while ensuring conflict-free 4DTs at an acceptable risk level α.

7.3 Framework of Chance-Constrained UTFM Optimization

7.3.1 Optimization Workflow

To optimize the disrupted 4D trajectories due to flight delays and probabilistic WOT uncertainties, we propose a chance-constrained UTFM optimization framework, shown in Fig. 7.4. The framework includes (1) modeling of disrupted 4DTs due to flight delays and WOT uncertainties, (2) formulation of a chance-constrained optimization model for the recovery of the disrupted 4DTs, (3) construction of a data-driven uncertainty set for estimating the probabilistic WOT, and (4) development of a hierarchical stochastic search (HSS) algorithm to solve the proposed optimization model.

The input of the framework is initial conflict-free 4DTs that have been optimized in the strategic phase. However, the uncertainties of departure delays and probabilistic WOT may disrupt flight plans and generate new conflicts. The first module of the framework is therefore designed to detect if there are flight delays and conflicts caused by these uncertainties. If no flight delays and conflicts are detected, the framework will output the initial conflict-free 4DTs. While, if flight delays are detected, the 4DTs of the delayed flights need to be checked with other non-delayed ones to detect any further conflicts. If there are conflicts, the 4DTs of the delayed flights need to be adjusted to resolve the conflicts. Note that flight conflicts also need to be detected even if there are no flight delays, as the probabilistic WOT of crossing waypoints may violate the deterministic separation requirement given in the initial 4DTs.

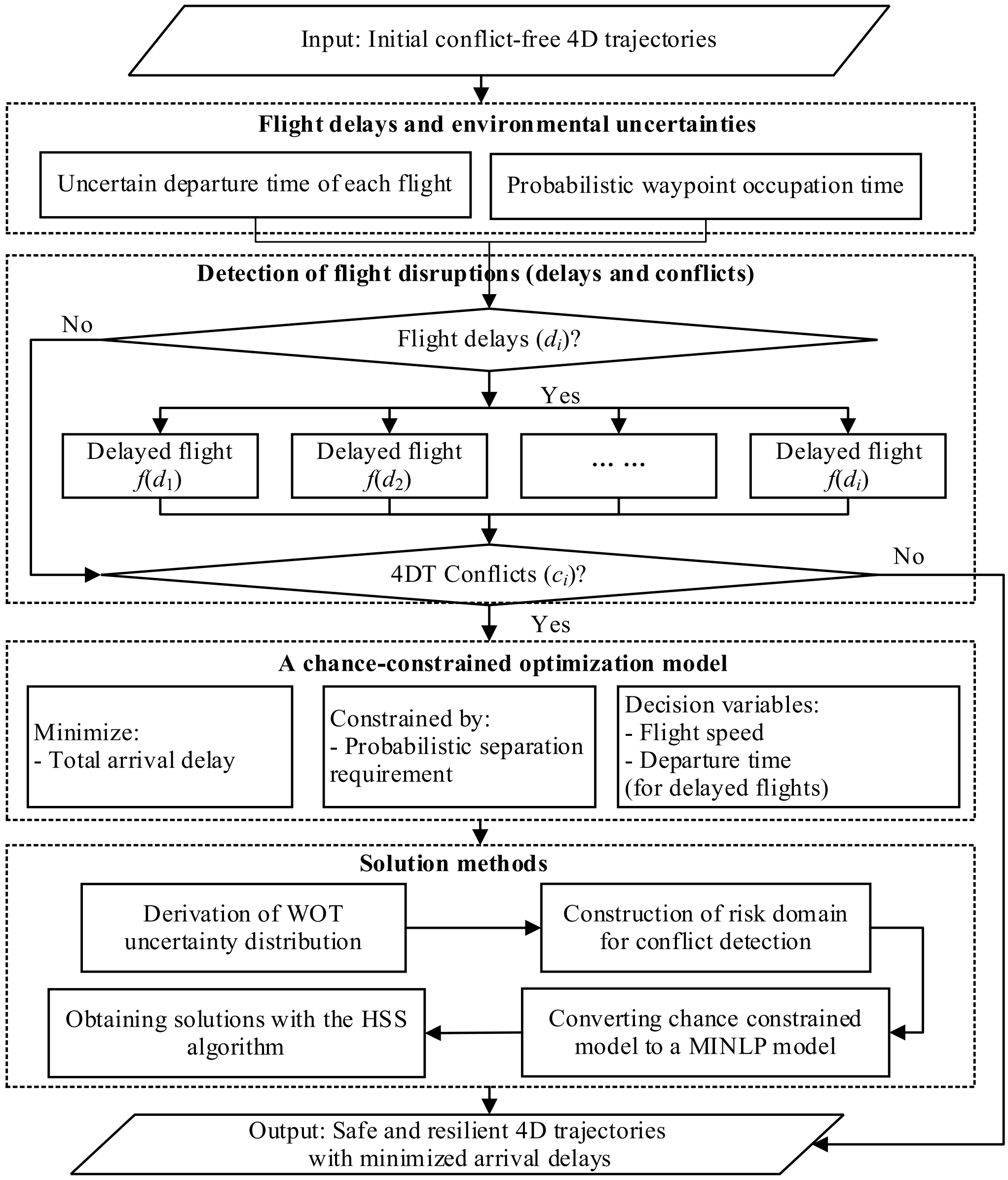


Fig. 7.4: Chance-constrained UTFM optimization framework for 4D trajectory optimization in the pre-tactical phase.

To optimize the disrupted flight trajectories in the presence of uncertainties, a chance-constrained UTFM optimization model is proposed with a probabilistic constraint of separation requirement at an acceptable risk level. The objective of the

UTFM model is to minimize total arrival delays with constraints of probabilistic separation requirement and flight duration. The decision variables of the model are flight speed and the actual departure time of delayed flights. The uncertain distribution of the WOT will be derived from historical flight experiment data via a data-driven approach. With the uncertainty distribution, the chance-constrained UTFM model can be then converted into a mixed integer nonlinear programming (MINLP) model by transferring the chance constraint into a deterministic constraint. The converted MINLP model will be solved by a developed hierarchical stochastic search (HSS) algorithm in a short time to meet the requirement of fast computation in the pre-tactical phase. The output after optimization will be safe and resilient 4D trajectories with minimized total arrival delays.

7.3.2 Deterministic UTFM Optimization Model

Pre-tactical UTFM optimization, forming another safety assurance layer, is to fill the gap between strategic deconfliction and tactical collision avoidance, with the aim of solving observed disrupted 4DTs (reactive approach) and improving their resilience in responding to future uncertainties (proactive approach). The objective of the model is to minimize the total arrival delay, and the decision variables are flight speed and actual departure time of all delayed flights. We first formulate the deterministic form of the UTFM optimization model and then develop the chance constraint of the probabilistic WOT.

Decision variables

$t'^{\text{dep}}_{f_k}$: actual departure time of f_k, a continuous variable.

$v_{s_{ij}\_f_k}$: flight speed of f_k at a segment s_{ij}, an integer variable.

Note that, the motivation of taking flight speed as an integer variable is due to safety and practical considerations for UAM operations [141], [205]. Using integer values for flight speed makes communication clearer between UAM operators and air traffic controllers to avoid confusion. It also allows better tracking and monitoring to detect speed deviations. Besides, the optimized speed by the proposed model is eventually used for manual or autonomous implementations where integer speed is a default setting in the current flight management system [211].

Objective function

The objective is to minimize the total arrival delay, which consists of ground delay determined by the postponed time of delay (Definition 1.1) and airborne delay defined as the actual airborne time (Definition 1.2). Based on Eqs. (7.1)-(7.4), the overall objective function is denoted as

$$\begin{aligned} T_{\text{delay}} &= \sum_{f_k \in F_D} t_{f_k}^{\text{arr\_dly}} = \sum_{f_k \in F_D} t'^{\text{arr}}_{f_k} - t_{f_k}^{\text{arr}} \\ &= \sum_{f_k \in F_D} \left(t'^{\text{dep}}_{f_k} + t_{f_k}^{\text{AAT}} \right) - t_{f_k}^{\text{arr}} \\ \Rightarrow T_{\text{delay}} &= \omega_{\text{ppn}} \sum_{f_k \in F_D} t'^{\text{dep}}_{f_k} + \omega_{\text{AAT}} \sum_{f_k \in F_D} \sum_{s_{ij\_f_k} \in S_{f_k}} \frac{l_{s_{ij}}}{v_{s_{ij\_f_k}}} \end{aligned} \tag{7.7}$$

where T_{Delay} is the total arrival delay, and F_D is the set of all delayed flights. Note that ω_{ppn} and ω_{AAT} are weightage factors for the postponed delay and the actual airborne time with $\omega_{\text{ppn}} + \omega_{\text{AAT}} = 1$, $\omega_{\text{ppn}}, \omega_{\text{ATT}} \in [0,1]$. The actual departure time of a delayed flight may need to be further postponed, as the 4DTs of the delayed flights may not be feasible due to congestion and conflicts. The initial arrival time $t_{f_k}^{\text{arr}}$ is a determined value and is not included in the final form of the objective function.

Constraints

The separation constraint is to ensure conflict-free 4D trajectories. The actual time separation $t_{w_{\text{cp}(i)}}^{\text{actual\_sep}}$ between two flights (ETA of a host flight $t_{w_{\text{cp}(i)\_f_k}}^{\text{ETA}}$ and WOT of another flight $t_{w_{\text{cp}(i)}}^{\text{WOT}}$ in the traffic) at a crossing waypoint $w_{\text{cp}(i)}$ should not be less than the separation requirement t_{sep}, denoted as

$$t_{w_{\text{cp}(i)}}^{\text{actual\_sep}} = \left| t_{w_{\text{cp}(i)\_f_k}}^{\text{ETA}} - t_{w_{\text{cp}(i)}}^{\text{WOT}} \right| \geq t_{\text{sep}} \tag{7.8}$$

where t_{sep} is a determined separation requirement in a deterministic conflict detection model.

$$t_{w_{\text{cp}(i)\_f_k}}^{\text{ETA}} = t'^{\text{dep}}_{f_k} + t_{w_{\text{cp}(i)\_f_k}}^{\text{AAT}} \tag{7.9}$$

Note that $t_{w_{\text{cp}(i)\_f_k}}^{\text{ETA}}$ can be obtained by actual departure time $t'^{\text{dep}}_{f_k}$ and actual airborne time $t_{w_{\text{cp}(i)\_f_k}}^{\text{AAT}}$. The term $t_{w_{\text{cp}(i)\_f_k}}^{\text{AAT}}$ is defined as the actual airborne time spent on each segment in the set $S_{w_{\text{cp}(i)\_f_k}}$ that contains segments from the origin to the current segment containing $w_{\text{cp}(i)}$, denoted as

$$t_{w_{\text{cp}(i)\_f_k}}^{\text{AAT}} = \sum_{s_{ij\_f_k} \in S_{w_{\text{cp}(i)\_f_k}}} \frac{l_{s_{ij}}}{v_{s_{ij\_f_k}}}, v_{ij\_f_k} \in \mathbb{Z}_+ \tag{7.10}$$

Flight duration constraint is to ensure the optimized trajectory can be completed within the power duration $t_{f_k}^{\mathrm{btry}}$ for a flight f_k, especially for battery-propelled eVTOLs.

$$t_{f_k}^{\mathrm{AAT}} = \sum_{s_{ij\_f_k} \in S_{f_k}} \frac{l_{s_{ij}}}{v_{s_{ij\_f_k}}} \leq t_{f_k}^{\mathrm{btry}} \tag{7.11}$$

Flight speed constraint is to ensure the planned speed is implementable for eVTOLs based on flight performances, denoted as

$$v_{f_k}^{\min} \leq v_{s_{ij\_f_k}} \leq v_{f_k}^{\max} \tag{7.12}$$

Note that $v_{f_k}^{\min}$ and $v_{f_k}^{\max}$ are minimum and maximum speeds for a flight f_k, and $v_{f_k}^{\min}$ may not start from zero as eVTOLs refer to both quadrotor and fixed-wing aerial vehicles.

The actual departure time $t'^{\mathrm{dep}}_{f_k}$ of a delayed flight will not be less than the initial departure time $t_{f_k}^{\mathrm{dep}}$, denoted as

$$t'^{\mathrm{dep}}_{f_k} \geq t_{f_k}^{\mathrm{dep}}, \forall t_{f_k}^{\mathrm{dep}} \in R_+ \tag{7.13}$$

7.3.3 Chance-Constrained Model

To develop the chance-constrained UTFM model, this work proposes to incorporate a probabilistic constraint for the separation requirement. Considering a probabilistic separation requirement $t_{w_{\mathrm{cp}(i)}}^{\mathrm{rqd\_sep}}$ between the ETA and WOT at a crossing waypoint $w_{\mathrm{cp}(i)}$, the probabilistic constraint for the loss of separation at a risk level $\alpha \in [0,1]$ can be denoted as

$$\mathrm{P_r}\left(t_{w_{\mathrm{cp}(i)}}^{\mathrm{actual\_sep}} < t_{w_{\mathrm{cp}(i)}}^{\mathrm{rqd\_sep}}\right) < \alpha, \forall w_{\mathrm{cp}(i)} \in \mathbb{W}_{\mathrm{cp}} \tag{7.14}$$

The probability that the actual separation $t_{w_{\mathrm{cp}(i)}}^{\mathrm{actual\_sep}}$ is smaller than the required separation $t_{w_{\mathrm{cp}(i)}}^{\mathrm{rqd\_sep}}$ at $w_{\mathrm{cp}(i)}$ should be less than the risk level α.

The condition of safe separation with a probabilistic guarantee at risk level α can be then derived as

$$\mathrm{P_r}\left(t_{w_{\mathrm{cp}(i)}}^{\mathrm{actual\_sep}} \geq t_{w_{\mathrm{cp}(i)}}^{\mathrm{rqd\_sep}}\right) \geq 1 - \alpha, \forall w_{\mathrm{cp}(i)} \in \mathbb{W}_{\mathrm{cp}} \tag{7.15}$$

Based on Eqs. (7.4), (7.8) and (7.9), the actual separation can be obtained as $t_{w_{\mathrm{cp}(i)}}^{\mathrm{actual\_sep}} = \left| t'^{\mathrm{dep}}_{f_k} + t_{w_{\mathrm{cp}(i)\_f_k}}^{\mathrm{AAT}} - t_{w_{\mathrm{cp}(i)}}^{\mathrm{WOT}} \right|$.

With the chance constraint of separation requirement, the chance-constrained UTFM optimization model is formulated in a probabilistic form, denoted as

$$\min:\ T_{\text{delay}} = \sum_{f_k \in F_D} \omega_{\text{ppn}} t'^{\text{dep}}_{f_k} + \omega_{\text{AAT}} t^{\text{AAT}}_{f_k} \tag{7.16}$$

Subject to:

$$\mathrm{P_r}\left(\left| t'^{\text{dep}}_{f_k} + t^{\text{AAT}}_{w_{\text{cp}(i)}\_f_k} - t^{\text{WOT}}_{w_{\text{cp}(i)}} \right| \geq t^{\text{rqd\_sep}}_{w_{\text{cp}(i)}} \right) \geq 1 - \alpha \tag{7.17}$$

$$t^{\text{AAT}}_{f_k} = \sum_{s_{ij}\_f_k \in S_{f_k}} \frac{l_{s_{ij}}}{v_{s_{ij}\_f_k}} \leq t^{\text{btry}}_{f_k} \tag{7.18}$$

$$v^{\min}_{f_k} \leq v_{s_{ij}\_f_k} \leq v^{\max}_{f_k} \tag{7.19}$$

$$t'^{\text{dep}}_{f_k} \geq t^{\text{dep}}_{f_k} \tag{7.20}$$

Compared with the deterministic model, the only difference comes from the replacement of the separation Constraint (7.8) with the Chance Constraint (7.17). That forms a chance-constrained UTFM optimization problem.

7.4 Solution Approach for Chance-Constrained UTFM Problem

The chance-constrained optimization model is an intractable problem, which is difficult to be solved directly [237]. Converting the chance-constrained problem into a standard optimization model makes it tractable, which has been proven effective for extensive chance-constrained optimization problems such as air traffic flow [237], risk-bounded path planning [250], and robot collision avoidance [263]. However, a successful conversion requires two key points. One point is to define an appropriate risk domain (or success set) for the random variables in chance constraints [250]. The other point, a more crucial one, is to obtain an accurate probabilistic description of the random variables in the optimization model [264]. However, in many real-life applications, we do not have information to obtain the accurate probability distribution of the randomness, not even for the estimated one (e.g., the lack of historical UAM operational data to estimate WOT deviations). Without knowing the probability distribution, obtained results may not be feasible for implementation; or the solution is feasible but has a far greater cost than the truly optimal one [264]. For instance, an underestimated (small) WOT may lead to a significant number of conflicts, because other flights may falsely find it safe and proceed to the waypoint while it is still occupied. On the contrary, an overestimated (large) WOT may lead to a significant number of delays, as the other flights are refused to enter the waypoint during the WOT even if it has not been occupied. To estimate the probability distribution on a finer scale, recent studies [241], [242], [264] proposed data-driven optimization frameworks, which use observation data of random variables to construct the uncertainty distribution. That is well-suited to the problem this Chapter studies.

To construct the data-driven uncertainty distribution, we conducted a number of flight experiments to obtain the historical flight trajectory data. The obtained data is used to derive the trajectory deviations on a temporal scale. By comparing with the pre-planned waypoint occupation time (time of a flight passing the waypoint), we can obtain the probability distribution of the WOT. By adjusting the risk level in the defined risk domain, the optimization model connects the user's risk preferences with the choice of uncertainty distribution.

7.4.1 Development of Data-Driven Uncertainty Distribution

In this section, we construct the uncertainty distribution of the probabilistic variables of $t_{w_{\mathrm{cp}(i)}}^{\mathrm{WOT}}$ and $t_{w_{\mathrm{cp}(i)}}^{\mathrm{prob\_ETA}}$. The two variables follow the same distribution, as the WOT at $w_{\mathrm{cp}(i)}$ is the ETA of another flight that is passing $w_{\mathrm{cp}(i)}$. The probabilistic $t_{w_{\mathrm{cp}(i)}}^{\mathrm{prob\_ETA}}$ consists of a pre-planned ETA $q_{w_{\mathrm{cp}(i)}}^{\mathrm{det\_ETA}}$ and an uncertainty u_t at a crossing waypoint $w_{\mathrm{cp}(i)}$. As $q_{w_{\mathrm{cp}(i)}}^{\mathrm{det\_ETA}}$ is a constant variable once the pre-planned 4D trajectory is generated, the probabilistic property of $t_{w_{\mathrm{cp}(i)}}^{\mathrm{prob\_ETA}}$ is only contributed by the uncertainty u_t that acts on the arrival time. The probabilistic $t_{w_{\mathrm{cp}(i)}}^{\mathrm{prob\_ETA}}$ can be denoted as

$$t_{w_{\mathrm{cp}(i)}}^{\mathrm{prob\_ETA}} = q_{w_{\mathrm{cp}(i)}}^{\mathrm{det\_ETA}} + u_t \tag{7.21}$$

where u_t is the uncertainty set that presents the ETA deviation between the stochastic ETA and the pre-planned ETA. Here u_t is a time-dependent set as the deviation may propagate during flight [265]. Note that $t_{w_{\mathrm{cp}(i)}}^{\mathrm{prob\_ETA}}$ is a time-varying variable.

To model the time-varying deviation uncertainty distribution of u_t, a data-driven approach is proposed with the use of flight experiment data. According to existing studies [117], [233], the ETA deviations tend to follow a Gaussian distribution. The uncertainty set, therefore, obeys the Gaussian distribution $u_t \sim \mathcal{N}(\mu_{\mathrm{dev}\_t}, \sigma_{\mathrm{dev}\_t}^2)$ where

$\mu_{\text{dev\_}t}$ and $\sigma^2_{\text{dev\_}t}$ are mean and variance of the uncertainty set at time t. The population mean and variance ($\mu_{\text{dev\_}t}$ and $\sigma^2_{\text{dev\_}t}$) can be considered equal to the observed sample ones $\hat{\mu}_{\text{dev\_}t}$ and $\hat{\sigma}^2_{\text{dev\_}t}$, if the observed sample size m is large enough (i.e. m >30) based on the Central Limited Theorem in statistical analysis.

We define $d_{\rho\_t} \in \mathbb{R}$ is the ETA deviation of ρth observation at time t, which is the difference between $\tau^{\text{ATA}}_{\rho\_t}$ and $\tau^{\text{ETA}}_{\rho\_t}$. Note that $\tau^{\text{ATA}}_{\rho\_t}$ is the actual time of arrival (ATA) at the observation location at time t, which is derived from historical flight trajectory data. While $\tau^{\text{ETA}}_{\rho\_t}$ is the ETA at the same observation location, which is obtained from initial flight plans.

The time-varying ETA uncertainty distribution of $u_t \sim \mathcal{N}(\mu_{\text{dev\_}t}, \sigma^2_{\text{dev\_}t})$ is modeled as

$$d_{\rho_t} = \tau^{\text{ATA}}_{\rho\_t} - \tau^{\text{ETA}}_{\rho_t}, \forall d_{\rho_t} \in R \tag{7.22}$$

$$\mu_{\text{dev\_}t} = \hat{\mu}_{\text{dev\_}t} = \frac{1}{m}\left(\sum_{\rho=1}^{m} d_{\rho_t}\right), m \in \mathbb{Z}_+ \tag{7.23}$$

$$\sigma^2_{\text{dev\_}t} = \hat{\sigma}^2_{\text{dev\_}t} = \frac{1}{m}\left(\sum_{\rho=1}^{m} (d_{\rho_t} - \hat{\mu}_{\text{dev\_}t})^2\right), m \in \mathbb{Z}_+ \tag{7.24}$$

According to Eq. (7.21), $q^{\text{det\_ETA}}_{w_{\text{cp}(i)}}$ would be a constant once the pre-planned 4DT is generated. As the uncertainty of the trajectory time deviation $u_t \sim \mathcal{N}(\mu_{\text{dev\_}t}, \sigma^2_{\text{dev\_}t})$, the probabilistic ETA also follows a normal distribution with $t^{\text{prob\_ETA}}_{w_{\text{cp}(i)}} \sim \mathcal{N}(\mu_t, \sigma^2_t)$, $\mu_t = \mu_{\text{dev\_}t} + q^{\text{det\_ETA}}_{w_{\text{cp}(i)}}$ and $\sigma^2_t = \sigma^2_{\text{dev\_}t}$. As the WOT is the ETA of the other flight in a conflict pair which comes first and occupies the waypoint, the WOT also follows the distribution of $t^{\text{WOT}}_{w_{\text{cp}(i)}} \sim \mathcal{N}(\mu_t, \sigma^2_t)$.

7.4.2 Risk Domain with Uncertainty Distribution

In this work, the risk domain is defined as the probabilistic waypoint occupation time $t^{\text{WOT}}_{w_{\text{cp}(i)}} \sim \mathcal{N}(\mu_t, \sigma^2_t)$ at a crossing waypoint $w_{\text{cp}(i)}$, which is caused by the early or late arrivals of a flight due to trajectory uncertainty u_t. Within a WOT (risk domain), other flights are not allowed to enter as one flight is passing this waypoint. The range of the WOT is determined by the acceptable risk level α. A higher risk level corresponds to a longer WOT, as it needs a larger safety buffer to handle higher uncertainties. A formal definition of the risk domain [250] based on risk level is given as follows.

Definition 3 (*Risk Domain*): A set $\mathcal{D} \in \mathbb{R}^d$ that satisfies

$$\mathrm{P_r}(\boldsymbol{T} \in \mathcal{D}) < \alpha \tag{7.25}$$

is called a risk domain at an acceptable risk level α for a random variable $\boldsymbol{T}$. Let $\boldsymbol{T}$ be a random variable with d-dimension and follows a d-dimension Gaussian distribution $\boldsymbol{T} \sim \mathcal{N}_d(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. In this work, the random variable is $t^{\mathrm{WOT}}_{w_{\mathrm{cp}(i)}}$, which follows 1-dimension Gaussian distribution (i.e., $d = 1$). The probability density function of $\boldsymbol{T}$ can be denoted as

$$f(\boldsymbol{T}) = \frac{1}{(2\pi)^{1/2}\sqrt{|\boldsymbol{\Sigma}|}} \exp\left[-\frac{1}{2}(\boldsymbol{T} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{T} - \boldsymbol{\mu})\right] \tag{7.26}$$

where $\boldsymbol{\mu}$ is the expectation of $\boldsymbol{T}$, and $\boldsymbol{\Sigma}$ is the covariance. As $d = 1$, the covariance of $\boldsymbol{T}$ equals to its variance $\boldsymbol{\Sigma} = \sigma^2$.

Given a risk level α (or confidence level $1 - \alpha$), we can find a value $t_{(1-\alpha)} = F^{-1}(1 - \alpha)$ such that

$$\mathrm{P_r}(\boldsymbol{T} < t_{(1-\alpha)}) = \mathrm{P_r}(\boldsymbol{T} < F^{-1}(1 - \alpha)) = 1 - \alpha \tag{7.27}$$

With the definitions above, we formulate the risk domain at a risk level α for the waypoint occupation time (WOT). Specifically, the probabilistic waypoint occupation time $\boldsymbol{t}^{\mathrm{WOT}}_{w_{\mathrm{cp}(i)}}$ is a one-dimension vector pertaining to a probabilistic period in which a crossing waypoint is occupied. Together with Eqs. (7.25)-(7.27), we can find a risk domain $\mathcal{D}_{w_{\mathrm{cp}(i)\_t}}$ at a risk level α for the crossing waypoint $w_{\mathrm{cp}(i)}$, formulated as

$$\mathcal{D}_{w_{\mathrm{cp}(i)\_t}} := \boldsymbol{t}^{\mathrm{WOT}}_{w_{\mathrm{cp}(i)}} < F^{-1}(1 - \alpha),\ \ \boldsymbol{t}^{\mathrm{WOT}}_{w_{\mathrm{cp}(i)}} \in \left[t^{\mathrm{WOT\_lower}}_{w_{\mathrm{cp}(i)}}, t^{\mathrm{WOT\_upper}}_{w_{\mathrm{cp}(i)}}\right] \tag{7.28}$$

The boundary of the risk domain $\mathcal{D}_{w_{\mathrm{cp}(i)\_t}}$ can be interpreted as a set of a period of time. The lower bound $t^{\mathrm{WOT\_lower}}_{w_{\mathrm{cp}(i)}}$ and upper bound $t^{\mathrm{WOT\_upper}}_{w_{\mathrm{cp}(i)}}$ are determined by the time-dependent normal distribution $\mathcal{N}_d(\mu_t, \sigma_t^2)$ at a risk level α. Where $\mu_t = \mu_{\mathrm{dev}\_t} + q^{\mathrm{det\_ETA}}_{w_{\mathrm{cp}(i)}}$ and $\sigma_t = \sigma_{\mathrm{dev}\_t}$ denote the expectation and standard deviation of the random WOT distribution at a crossing waypoint $w_{\mathrm{cp}(i)}$ at time t. To ensure that the chance of conflict is less than the acceptable risk level α, the ETA of a flight f_k on a crossing waypoint $t^{\mathrm{ETA}}_{w_{\mathrm{cp}(i)\_f_k}}$ should not be inside the WOT defined by its risk domain $\mathcal{D}_{w_{\mathrm{cp}(i)\_t}}$. According to Eq. (7.17), the separation constraint can be denoted as

$$\mathrm{P_r}\left(\left|t^{\mathrm{ETA}}_{w_{\mathrm{cp}(i)\_f_k}} - \boldsymbol{t}^{\mathrm{WOT}}_{w_{\mathrm{cp}(i)}}\right| \geq t^{\mathrm{rqd\_sep}}_{w_{\mathrm{cp}(i)}}\right) \geq 1 - \alpha \tag{7.29}$$

Either $t^{\mathrm{ETA}}_{w_{\mathrm{cp}(i)\_f_k}} \geq t^{\mathrm{WOT\_upper}}_{w_{\mathrm{cp}(i)}}$, meaning that a flight arrives at the waypoint after WOT, then

$$\begin{aligned} &\mathrm{P_r}\left(\boldsymbol{t}^{\mathrm{WOT}}_{w_{\mathrm{cp}(i)}} \leq t^{\mathrm{ETA}}_{w_{\mathrm{cp}(i)\_f_k}} - t^{\mathrm{rqd\_sep}}_{w_{\mathrm{cp}(i)}}\right) \geq 1-\alpha \\ \Rightarrow\ &\left(t^{\mathrm{ETA}}_{w_{\mathrm{cp}(i)\_f_k}} - t^{\mathrm{rqd\_sep}}_{w_{\mathrm{cp}(i)}}\right) \geq F^{-1}(1-\alpha) \end{aligned} \tag{7.30}$$

Or $t^{\mathrm{ETA}}_{w_{\mathrm{cp}(i)\_f_k}} < t^{\mathrm{WOT\_lower}}_{w_{\mathrm{cp}(i)}}$, meaning that a flight arrives at the waypoint before it has occupied, then

$$\begin{aligned} &\mathrm{P_r}\left(\boldsymbol{t}^{\mathrm{WOT}}_{w_{\mathrm{cp}(i)}} \geq t^{\mathrm{ETA}}_{w_{\mathrm{cp}(i)\_f_k}} + t^{\mathrm{rqd\_sep}}_{w_{\mathrm{cp}(i)}}\right) \geq 1-\alpha \\ \Rightarrow 1 - &\mathrm{P_r}\left(\boldsymbol{t}^{\mathrm{WOT}}_{w_{\mathrm{cp}(i)}} < t^{\mathrm{ETA}}_{w_{\mathrm{cp}(i)\_f_k}} + t^{\mathrm{rqd\_sep}}_{w_{\mathrm{cp}(i)}}\right) \geq 1-\alpha \\ \Rightarrow\ &\mathrm{P_r}\left(\boldsymbol{t}^{\mathrm{WOT}}_{w_{\mathrm{cp}(i)}} < t^{\mathrm{ETA}}_{w_{\mathrm{cp}(i)\_f_k}} + t^{\mathrm{rqd\_sep}}_{w_{\mathrm{cp}(i)}}\right) \leq \alpha \\ \Rightarrow\ &\left(t^{\mathrm{ETA}}_{w_{\mathrm{cp}(i)\_f_k}} + t^{\mathrm{rqd\_sep}}_{w_{\mathrm{cp}(i)}}\right) \leq F^{-1}(\alpha) \end{aligned} \tag{7.31}$$

where $t^{\mathrm{ETA}}_{w_{\mathrm{cp}(i)\_f_k}} = t'^{\mathrm{dep}}_{f_k} + t^{\mathrm{AAT}}_{w_{\mathrm{cp}(i)\_f_k}}$.

7.4.3 Conversion of Chance-constrained Model to MINLP Model

Compared with the probabilistic constraint in Eq. (7.17), Constraints (7.30) and (7.31) are deterministic functions, which are tractable and implementable to our optimization problem. As a flight arrives at a waypoint either after (Eq. (7.30)) or before (Eq. (7.31)) it has been occupied by another flight, we convert this logical condition into linear constraints by introducing binary variables ξ_1, ξ_2 and a sufficiently large number M, which is always greater than $t'^{\mathrm{dep}}_{f_k} + t^{\mathrm{AAT}}_{w_{\mathrm{cp}(i)\_f_k}} + t^{\mathrm{rqd\_sep}}_{w_{\mathrm{cp}(i)}}$. With that, the chance-constrained UTFM optimization model can be converted as a deterministic optimization model, denoted as

$$\min: T_{\mathrm{delay}} = \omega_{\mathrm{ppn}} \sum_{f_k \in F_D} t'^{\mathrm{dep}}_{f_k} + \omega_{\mathrm{AAT}} \sum_{f_k \in F_D} \sum_{s_{ij\_f_k} \in S_{f_k}} \frac{l_{s_{ij}}}{v_{s_{ij\_f_k}}} \tag{7.32}$$

Subject to:

$$t'^{\mathrm{dep}}_{f_k} + t^{\mathrm{AAT}}_{w_{\mathrm{cp}(i)\_f_k}} - t^{\mathrm{rqd\_sep}}_{w_{\mathrm{cp}(i)}} \geq F^{-1}(1-\alpha) - M\xi_1 \tag{7.33}$$

$$t'^{\mathrm{dep}}_{f_k} + t^{\mathrm{AAT}}_{w_{\mathrm{cp}(i)\_f_k}} + t^{\mathrm{rqd\_sep}}_{w_{\mathrm{cp}(i)}} \leq F^{-1}(\alpha) + M\xi_2 \tag{7.34}$$

$$\xi_1 = \begin{cases} 0, \text{if } \left(t'^{\mathrm{dep}}_{f_k} + t^{\mathrm{AAT}}_{w_{\mathrm{cp}(i)\_f_k}}\right) \geq t^{\mathrm{WOT\_upper}}_{w_{\mathrm{cp}(i)}} \\ 1, \text{otherwise} \end{cases} \tag{7.35}$$

$$\xi_2 = \begin{cases} 0, \text{if } \left(t'^{\text{dep}}_{f_k} + t^{\text{AAT}}_{w_{\text{cp}(i)}\_f_k}\right) < t^{\text{WOT\_lower}}_{w_{\text{cp}(i)}} \\ 1, \text{otherwise} \end{cases} \tag{7.36}$$

$$\xi_1 + \xi_2 = 1 \tag{7.37}$$

$$t^{\text{AAT}}_{f_k} = \sum_{s_{ij\_f_k} \in S_{f_k}} \frac{l_{s_{ij}}}{v_{s_{ij\_f_k}}} \leq t^{\text{btry}}_{f_k} \tag{7.38}$$

$$v^{\min}_{f_k} \leq v_{s_{ij_{f_k}}} \leq v^{\max}_{f_k} \tag{7.39}$$

$$t'^{\text{dep}}_{f_k} \geq t^{\text{dep}}_{f_k} \tag{7.40}$$

The converted deterministic model is a mixed integer nonlinear programming (MINLP) model, which has two decision variables of the continuous $t'^{\text{dep}}_{f_k}$ and the integer $v_{s_{ij\_f_k}}$ under both the linear Constraints (7.33)-(7.34), (7.39)-(7.40), 0-1 integer constraints (7.35)-(7.36), and nonlinear Constraints (7.38).

7.4.4 Hierarchical Stochastic Search (HSS) Algorithm

Solutions for solving the NP-hard MINLP problems have been widely studied. Exact methods such as branch-and-bound or cutting-plane-based algorithms [120] are always applied to a relaxed form of MINLP to solve linear programming relaxations tightened by cutting planes at each node [266]. However, these methods would become exhaustive without an effective estimation of the lower and upper bounds of branches in the search space, making these methods inapplicable to solve the problem within an acceptable time at the pre-tactical phase in this work. Commercial solvers can also solve MINLP problems. However, the performance of solvers would be significantly reduced if the objective function is not linear [267] such as the one (Eq. (7.32)) in this work. Solver has also been proven as a highly time-consuming way to solve the MINLP problem [237], which makes it an inappropriate one in this work. Furthermore, the performance of exact methods and solvers drops with the increase of the search space size, while metaheuristic algorithms are less sensitive to the space size and nonlinear objectives[222], which motivates the use of a metaheuristic method.

In this work, we develop a hierarchical stochastic search (HSS) algorithm to solve the modelled MINLP problem with fast computation. The HSS algorithm is inspired by a novel and powerful metaheuristic optimization framework [194], which transforms objectives and constraints into weighted penalties by adopting the constraint transcription method [268]. Specifically, weights of constraint penalties will be assigned with much higher orders of magnitude than objectives to prevent violations of constraints. The separation constraints (Eqs. (7.33)-(7.37)), safety-critical ones, are also transferred to a penalty. Output solutions could be infeasible because of separation violations. A hierarchical safety guarantee is introduced by increasing the weights of the separation constraint in the next iterations if the separation is violated.

After transformation, the developed MINLP model is converted as a cost function, formulated as

$$\begin{cases} h_{\text{cost}} = C_{\text{obj}} + C_{\text{sep}} + C_{\text{btry}} \\ C_{\text{obj}} = T_{\text{delay}} \\ C_{\text{sep}} = \omega_{\text{sep}} n_{\text{conflict}} \\ C_{\text{btry}} = \omega_{\text{btry}} n_{\text{btry\_flat}} \end{cases} \tag{7.41}$$

where h_{cost} is the total cost, and C_{obj} is the cost of the objective function as shown in Eq. (7.32). Note that C_{sep} and C_{btry} are penalty costs of separation violation and battery duration violation, which are correlated with the number of flight conflicts n_{conflict} and number of flight duration violations $n_{\text{btry\_flat}}$, respectively. Here ω_{sep} and ω_{btry} are weights of conflict penalty and duration violation penalty. With the total cost function, the HSS algorithm can be used to find optimized results of the UTFM problem.

In the HSS algorithm, exploitation and exploration are two core processes [141]. The exploitation process is to utilize the current best information by creating new solution points that are adjacent to the current best solution point distributed by Gaussian random walks, which can be denoted as

$$P_{\text{new}}^{\eta} = P_{\text{curr}} + (\varepsilon P_{\text{curr}_{\text{best}}} - (1-\varepsilon) P_{\text{curr}}) + \text{Gau}(\mu_P, \sigma_P) \tag{7.42}$$

where P_{new}^{η} is the ηth new point diffused by the current point P_{curr} for exploitation. Note that $P_{\text{curr\_best}}$ is the best solution point in the current vector, and ε is the weightage factor to adjust the degree of exploitation. Here μ_P and σ_P are the mean and standard deviation of positions of the new points.

The exploration process is to ensure a high diversity of solutions by searching for other solution points that are not adjacent to the current best point, denoted as

$$\begin{cases} P_{\text{new}}^{\zeta} = P_{\text{curr}} - \varepsilon'(P'_{\text{rdm}} - P_{\text{curr\_best}}), \varepsilon' \leq 0.5 \\ P_{\text{new}}^{\zeta} = P_{\text{curr}} + \varepsilon'(P''_{\text{rdm}} - P'_{\text{rdm}}), \varepsilon' > 0.5 \end{cases} \tag{7.43}$$

where P_{new}^{ζ} is the ζth updated position of P_{new}^{η}, and ζ is the number of points being updated. Here P'_{rdm} and P''_{rdm} are two randomly generated points in the solution vector, and ε' is the weightage to adjust the degree of exploration. In this work, the factors of

ε and ε' are both taken as 0.5 for exploitation and exploration balancing. The detailed fundamental development of the stochastic optimization framework can be found in [141].

As output solutions may still be infeasible due to violations of the separation and battery duration. A hierarchical safety guarantee is introduced by increasing the penalty weights of separation constraint and battery duration if violations are detected. With more violations detected, the corresponding updated penalty weights ω'_{sep} and ω'_{btry} will be given a larger number. The weightage update function is given as

$$\begin{cases} \omega'_{\text{sep}} = \omega_{\text{sep}} \left(1 + \dfrac{n_{\text{conflict}}}{N_{F_D}}\right) \\ \omega'_{\text{btry}} = \omega_{\text{btry}} \left(1 + \dfrac{n_{\text{btry\_flat}}}{N_{F_D}}\right) \end{cases} \tag{7.44}$$

Here, N_{F_D} is the total number of delayed flights.

With the modelling above, the pseudo-code of the population-based hierarchical stochastic search (HSS) algorithm is presented in Algorithm 1.

Algorithm 1: Hierarchical stochastic search (HSS) algorithm

Input: Initial conflict- free 4DTs, Lower and upper bounds of decision variables (v, t_{dep})
Initialization: generate a population of solution vector P with N-dimension
Selection: obtain the initial best solution by evaluating the cost value (**Eq. (7.41)**)
while $n_{\text{conflict}} \neq 0$ | | $n_{\text{btry\_flat}} \neq 0$ % Optimization process
  begin: *Exploitation process*
    Searching neighbors of the initial best solution (**Eq. (7.42)**)
    Update the current best solution point ($P_{\text{curr\_best}}$) and cost value (h_{cost})
  end
  begin: *Exploration process*
    Explore solution points based on two random points in P (**Eq. (7.43)**)
    Update $P_{\text{curr\_best}}$ and h_{cost}
  end
  begin: *Evaluation process*
    [n_{conflict}, $n_{\text{btry\_flat}}$] $=f(P_{\text{curr\_best}})$ % obtain number of violations by **Eqs. (7.33)-(7.38)**
      if $n_{\text{conflict}} > 0$ and/or $n_{\text{btry\_flat}} > 0$
        Update weights of the conflict penalty and/or the duration violation penalty by **Eq. (7.44)**
      end
  end
Return (P_{best}, h_{cost})

In the initialization process, the candidate solution vectors are sampled based on the population size. Each solution vector has N-dimension, representing the decisions for all delayed flights N_{F_D}. The current best solution vector $P_{\text{curr\_best}}$ in each iteration is selected from the candidate solutions based on the cost function Eq. (7.41). The HSS algorithm iterates to search the near-optimal solution vectors till meet the stopping criterion defined by $n_{\text{conflict}} \neq 0 \ \| \ n_{\text{btry\_flat}} \neq 0$. Meaning that the iteration stops when the number of conflicts and duration violations are both zero.

The number of basic executions of the algorithm is n in the main loop (the While loop). Given a population size, the number of executions increases proportionally with the increase in the input size N_{F_D}, as the termination condition is guaranteed for all cases of input size by adjusting not only the flight speed but the departure time to meet the termination condition if the speed adjustment fails to do so. That makes the time complexity of the algorithm $O(n)$.

7.5 Numerical Analysis and Results

In this section, simulations are conducted to demonstrate the effectiveness, robustness, and scalability of the developed chance-constrained UTFM optimization model and the HSS algorithm. The uncertainty distribution is first established based on the flight experiment data using a specific drone [269]. The optimization model is then applied to several cases to solve the disrupted 4DTs due to delays and trajectory deviations. Sensitivity analysis is also performed in various environments with different UAM traffic patterns and densities to demonstrate the generalizability and scalability of the proposed model and algorithm. Computations in this section were conducted using Matlab on a computer with Intel i9 2.5GHz processors and 32 GB RAM.

7.5.1 Uncertainty Distribution based on Flight Experiment Data

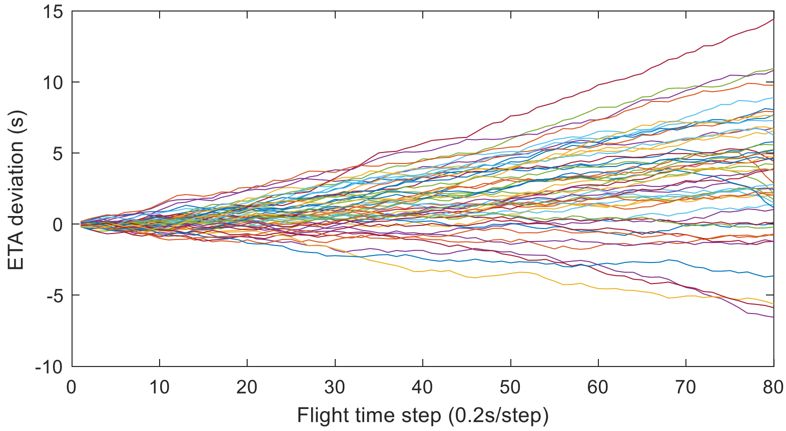
The flight experiments were conducted with the drone of DJI Inspire-1 (~3kg) under beyond visual line of sight (BVLOS) operations in the Waypoint Model using the DJI mobile application. The waypoints were manually chosen by setting (latitude, longitude, altitude), which formed a 3D quad pattern trajectory with a side length of 140 m. The flight speed was taken as 7 m/s, so the ETA to each waypoint can be computed by adding the expected airborne time with the departure time. With the 3D trajectory and ETA to each waypoint, the pre-planned 4D trajectories were obtained. After the experiment, a total of 52 independent flight trajectories with tracking information were obtained. As the scope of this work is on the ETA deviations, only the along-track trajectory deviation [144] is considered in the analysis. By comparing the along-track time deviations of the pre-planned 4DTs and the historical ones, the cumulative ETA deviations at each step (0.2 s/step, as telemetry data are recorded at 5 Hz) were obtained and presented in Fig. 7.5. Note that the ETA deviation is defined as the historical ETA minus the pre-planned ETA at each step (Eqs. (7.19)-(7.21)). The flight time step starts with the latest time of the trajectory location being calibrated.

The cumulative ETA deviations at each time step range from negative to positive values, as shown in Fig. 7.5(a). A negative deviation represents the historical ETA is smaller than the pre-planned one, meaning that the flight arrives earlier than the pre-planned ETA, while a positive deviation means the flight arrives late. In general, the ETA deviation increases with the increase of the time step to the last calibration. To investigate the type of deviation distribution, normality tests were performed for deviations at each step that has one sample group (ETA deviation) with a size of 52. Obtained results are presented in Fig. 7.5(b), and the ETA deviations at all time steps follow a normal distribution (p-value>0.05) at a 95% confidence level. The mean and variance of the normal distributions at each step are obtained and shown in Fig. 7.5(c).

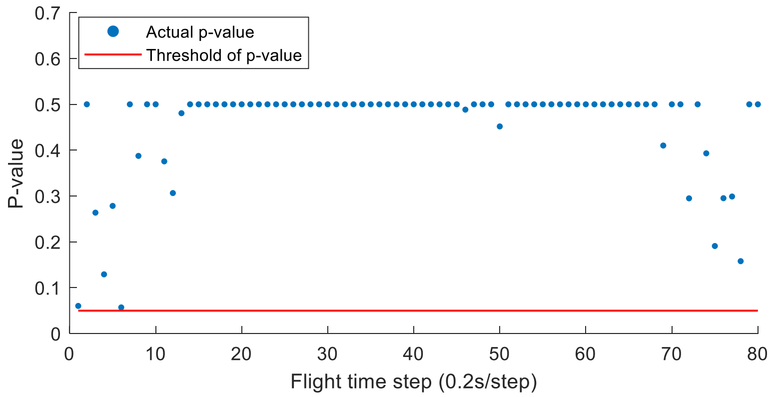
In this section, we demonstrate the feasibility of the assumption made in Section IV-A that the ETA uncertainty u_t follows a normal distribution $\mathcal{N}(\mu_{\mathrm{dev\_}t}, \sigma^2_{\mathrm{dev\_}t})$. With the obtained uncertainty distribution, the UTFM optimization model can be used to optimize the disrupted 4DTs.

7.5.2 Optimization Performances of UTFM Optimization Model

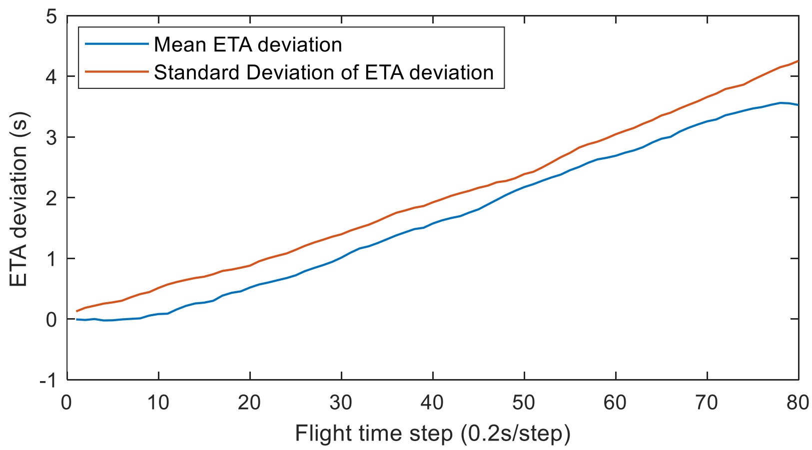
In this section, we demonstrate the performance of the proposed UTFM optimization model and HSS algorithm. The simulation environment is selected from an area in Singapore with a size of 6,000m×6,000m×120m (width×length×altitude). The selected airspace is meshed into standard 3D blocks with a size of 100m×100m×30m. A total of 100 flights are randomly generated in the airspace at a one-hour time window. The origin and destination of these flights are randomly chosen in the environment under the constraint that the flight distance is in the range of [2, 8] km, with a speed of 7 m/s in accordance with the flight experiments in Section V-A. The initial conflict-free 4DTs are generated by a previous study [141]. The percentage of delayed flights is taken as 20%, and delayed flights are randomly chosen from the 100 flights. The time of delays are random variables that follow a normal distribution, and the mean and standard deviation of the distribution are taken as 600 and 300, respectively [117] in this study.



(a) Cumulative ETA deviations at each step (0.2 s/step)



(b) Normality test results (p-value>0.05, normal)



(c) Mean and standard deviation of the ETA deviations

Fig. 7.5: Uncertainty distribution of ETA deviations.

Other parameters used in the simulations are given in Table 7.1. The decision variables are the actual departure time t'^{dep} and the flight speed v. Here t'^{dep} is in the range of [1, 3600] s in a one-hour time window, and the flight speed is taken as integers ranging from [5, 20] m/s. The lower bound of the speed is taken as 5 m/s, meaning that hovering is not an option, as our model considers both quad-rotor drones and fixed-wing eVTOLs. The upper bound of the speed (20 m/s) and battery duration t^{btry} (18 min) are constrained by the performance of DJI Inspire-1 [269] used in simulations. The risk level α is given as 0.05. The weightage factors of ω_{ppn} and ω_{AAT} are both taken as 0.5, indicating the same significance of the ground delay and the airborne delay. The initial penalty weights for conflict and duration violation are taken as 300, which represents that the penalty for one violation is equivalent to 300 seconds of delays in the optimization model. Uncertainty distribution of the ETA deviation is obtained in Section V-A. These uncertainties are ingested into the initial 4DTs, causing new conflicts as shown in Fig. 7.6(a). The UTFM optimization model is then applied to optimize the disrupted flights, and obtained results are shown in Fig. 7.6.

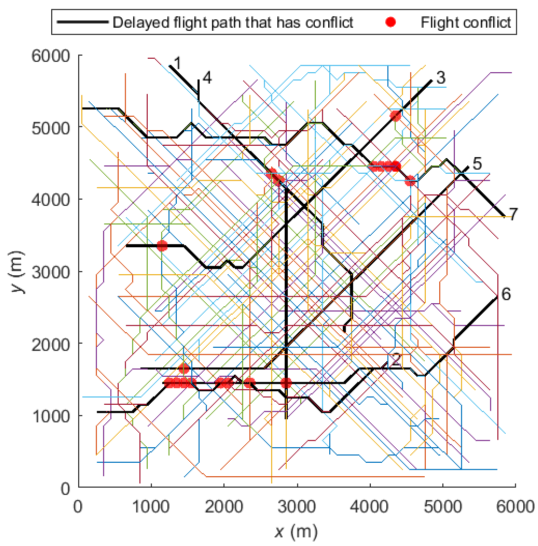
Among the 20 delayed flights, there are 7 flights with their 4DTs intersected with other flights yielding 26 new conflicts, shown in Fig. 7.6(a). It means that not all delayed flights will cause new conflicts if the delayed ETAs at crossing waypoints do not fall into the corresponding risk domain that is defined by the waypoint occupation time. The flights with conflicts are labeled from #1 to #7 at the origin position of their paths, and the correlations between ETA and WOT of these conflicts are demonstrated in Fig. 7.6(b).

Table 7.1: Parameter settings in the simulations.

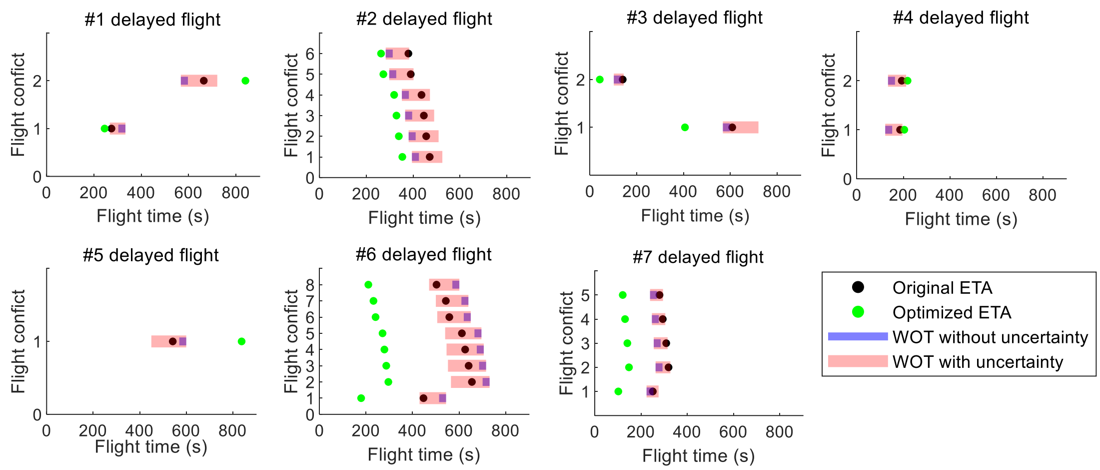
Types	Parameters	Range of values
Decision variables	v (m/s)	[5, 20]
	t'^{dep} (s)	[1, 3600]
Constraint thresholds	t^{btry}(min)	18
	α	0.05
Weights in objective and penalties	ω_{ppn}	0.5
	ω_{AAT}	0.5
	ω_{sep}	300
	ω_{btry}	300

The #1 delayed flight starts from the origin (x=1300, y=5900) to the destination (x=2900, y=1000), creating two conflicts at the flight time of 244 s and 664 s, shown at the top left of Fig. 7.6(b). The original ETAs (black-filled circles) of these two conflicts are outside the risk domain defined by a deterministic WOT (blue-shaded areas) if uncertainty is not considered. However, the original ETAs fall into the risk domain under the WOT with uncertainty (red-shaded areas) that is caused by ETA deviations, yielding flight conflicts. Note that with the increase of the flight time, the probabilistic WOT becomes larger due to the uncertainty propagation. For instance, the WOT at the second conflict position (on 664 s) is significantly larger than the one in the first position (on 244 s) for the #1 delayed flight. By taking the uncertainty at a risk level of 0.05, the optimized ETAs (green-filled circles) are all outside the risk domains, meaning successful resolutions for conflicts. The optimization performance also extends to the other six delayed flights, and all conflicts are solved with optimized ETAs that are cleared from the probabilistic WOTs.

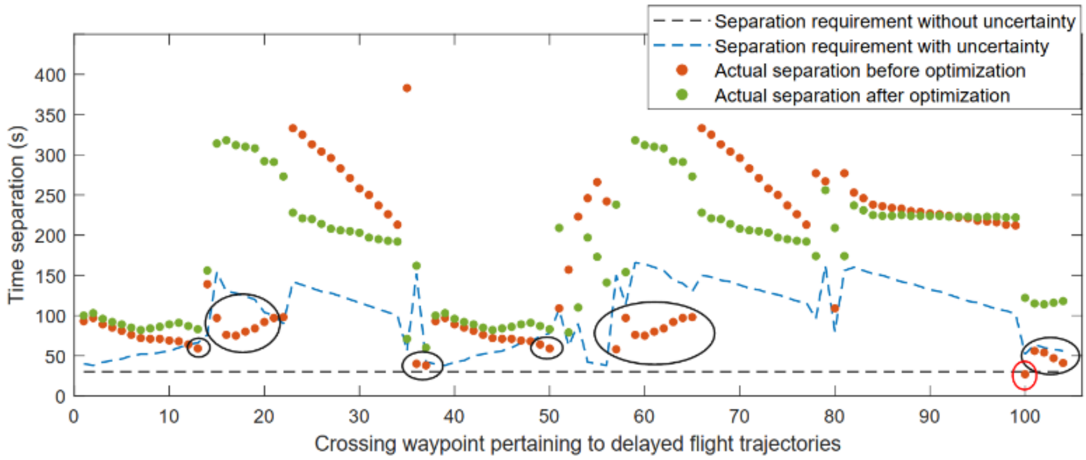
To have a deeper view of the conflict condition for each crossing waypoint pertaining to the delayed flights, the correlations between actual separation and separation requirement before and after optimization are presented in Fig. 7.6(c). Delayed flights have a total of 105 crossing waypoints in which conflicts may occur if the time separation is violated. Under a deterministic separation requirement (black dotted line), only one conflict is detected (marked by the red ellipsoid), resulting in 25 misdetections (marked by black ellipses) if consider the ETA deviation, an inevitable uncertainty in real-world UAM operations. After optimization, all separation violations are successfully resolved, as the optimized ETAs are outside the probabilistic WOTs that form the risk domains. Compared with the traditional 4DT optimization model which does not consider uncertainties, our proposed chance-constrained UTFM model significantly reduces and solves the misdetections of conflicts from 25 to zero at a risk level of 0.05.



(a) Conflicts caused due to flight delays and ETA deviations



(b) Conflict detection and resolution in conditions with and without WOT uncertainties



(c) Actual separation versus separation requirement in conditions with and without uncertainties

Fig. 7.6: Optimization results of the disrupted 4DTs due to flight delays and ETA uncertainty. Note that the optimized ETAs can be earlier or later than the original ETAs, as demonstrated by delayed flight #1 in (b). As departure time can only be delayed, earlier optimized ETAs can only be done by increasing flight speed. Also note that the optimized ETAs of the #4 delayed flight in (b) are not inside the WOT. Large plotted green circles are for easy reading.

7.5.3 Reliability Testing of Proposed Model and Algorithm

In Section V-B, a case with 20% delayed flights is performed with a total of 100 flights. More delayed flights represent more flight disruptions that need to be handled by the optimization model. Which increases the complexity of the optimization problem, as the input size of the problem increased. The increased input size brings challenges for computational efficiency and solution quality. To demonstrate how different percentages of delayed flights will impact the initial 4DTs and optimization performances, we conduct sensitivity analysis on the percentage of delayed flights ranging from [1, 50] % with an interval of one percent. Note that the increment of the delay percentage is obtained by adding all existing delayed flights with one percent of new delayed flights that are selected from the unchosen flight set. That makes a reliable comparison in scenarios with different delay percentages by reducing the randomness effects of mission diversities (e.g., different departure times and flight distances). To further demonstrate the reliability of the proposed optimization model, simulations for each percentage of delayed flights are conducted 50 times independently in different traffic patterns that are randomly generated. Obtained results are presented in Fig. 7.7.

In general, with the increase in the percentage of delayed flights, the degree of disruption on the initial conflict-free 4DTs becomes higher in terms of the number of flight conflicts, number of rejected flights, and average delay. Before optimization, the number of conflicts saw a slight increase (blue curve in Fig. 7.7(a)) within low delay percentages (<6%), followed by a significant increase with the rise of delayed flight percentage until reaching the threshold (18%) of a complete disruption where the number of conflicts has no notable increases. It reveals that within a low range of delay percentage, the disruptions to the entire traffic flow are not significant. That is because the chance is low to cause ETA clashes by only a few delayed flights. However, exceeding this low range, the number of conflicts increases rapidly, indicating that the initial conflict-free 4DTs are heavily impacted by the delayed flights, and the notable increase ceased until exceeding the threshold of a complete disruption where the effectiveness of the initially optimized conflict-free 4DTs has been degraded.

After optimization, all conflicts are resolved by the proposed optimization model. While it has sacrifices of an increased number of flight rejections (Fig. 7.7(b)) and increased average flight delays (Fig. 7.7(c)). As in the objective function (Eq. (7.32)), the departure time of flights would be postponed solving conflicts. If the postponed departure time of a flight exceeds the time window (one hour in this case), this flight will be rejected, meaning that the traffic density exceeds the airspace capacity. As shown in Fig. 7.7(b), the average number of rejected flights is zero in the 50 independent simulations when the percentage of delayed flights is below 16%. In that range, the optimization model can solve all conflicts without the need to reject any flights. Which could facilitate the development of the capacity threshold for the UAM traffic management system. Flights exceeding the capacity threshold should be allocated to next-time windows to avoid rejections. After optimization, the average flight delays are reduced in a low range of percentage (<9%), as shown in Fig. 7.7(c). When the percentage of delayed flights exceeds 10%, the average delay of optimized flights outnumbers the initial ones, which is the cost of solving conflicts. While the gaps of the delays before and after optimization are maintained in a linear manner.

We also conduct simulations to investigate the influence of ETA uncertainty on optimization performance, as ETA uncertainty causes difficulties in generating robust flight 4D trajectories. The control variable in the simulations is the mean ETA deviation, which is taken from the range of [1, 15] minutes with an interval of one minute. Simulations are performed under each ETA deviation condition and obtained results demonstrate that the proposed method can solve all flight conflicts in an increasing ETA uncertainty condition. However, similar to the results in Fig. 7.7(b), the proposed method has a cost with a slightly increased number of flight rejections.

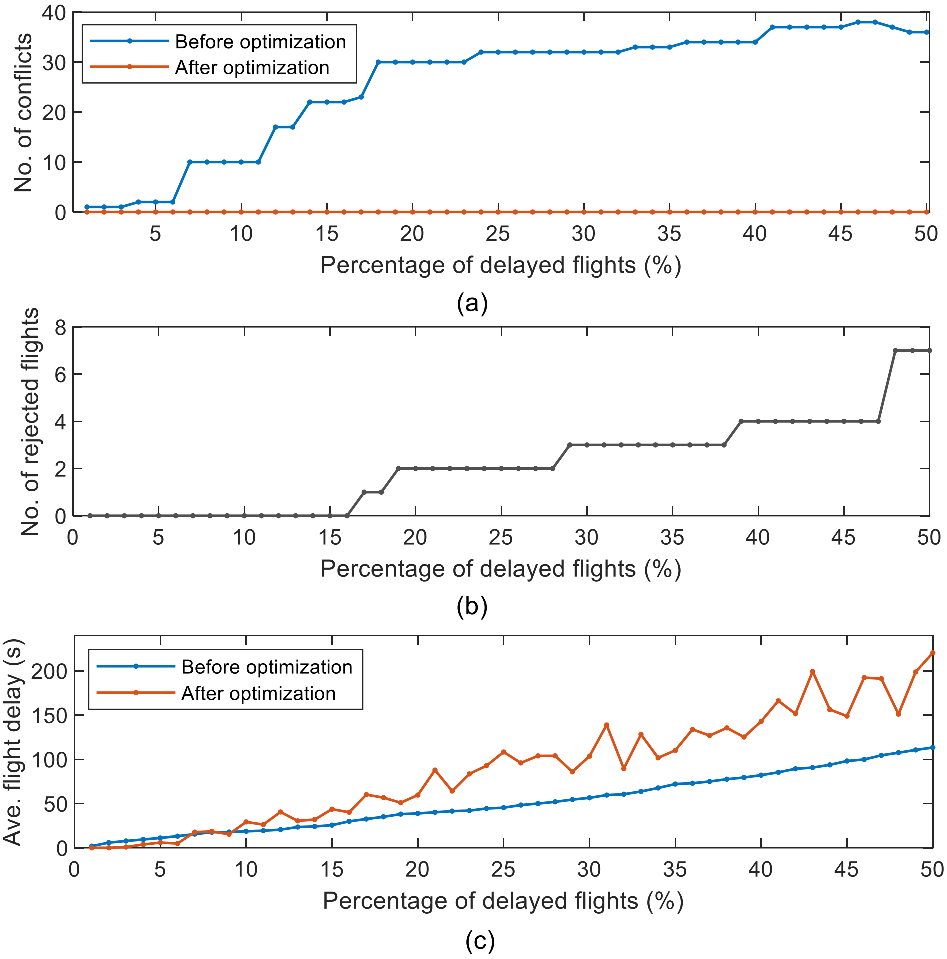


Fig. 7.7: Reliability of the proposed model in 50 different traffic patterns with an incremental percentage of delayed flights. (At 0.05 risk level)

7.5.4 Impacts on UTFM under Different Risk Levels

The simulations mentioned above are conducted at a risk level of 0.05. To demonstrate the influences of different risk levels on the UAM traffic and the corresponding optimization performances, simulations are conducted with six risk levels of {0.2, 0.1, 0.05, 0.025, 0.01, 0.001} with 20% delayed flights in 100 flights. Obtained results are presented in Table 7.2.

With the decrease in the risk level, all indicators become worse. Because a lower risk level represents a more stringent safety requirement, which is equivalent to a higher separation requirement. A higher separation requirement results in a smaller airspace capacity, which yields more conflicts and delays if the traffic density remains unchanged. That is why the number of conflicts created by uncertainties (2nd row in Table 7.2) in the initial 4DTs saw an exponential increase with the decrease of risk levels. The optimized results show a similar pattern with all conflicts resolved (3rd row) while at little cost of flight rejections (4th row) and delays (5th row).

Table 7.2: Optimization performances under different risk levels.

Indicators\Risk level (α)		0.2	0.1	0.05	0.025	0.01	0.001
Initial 4DTs	No. of conflicts caused	22	24	30	35	39	43
After Optimization	No. of conflicts	0	0	0	0	0	0
	No. of rejected flights	0	1	1	1	1	1
	Ave. delay change (s)	24.0	21.4	22.3	25.7	32.5	47.0

7.5.5 Scalability Analysis

In the simulations above, the optimization model performs well in solving flight conflicts and optimizing delays. The reliability testing also demonstrates promising results in different traffic patterns at different risk levels, with a traffic density of 100 (flights/hour/36 km^2). However, higher traffic density could lead to more conflicts and leave less space for resolution maneuvers. To investigate the performances of the proposed model and algorithm in scenarios with different traffic densities, scalability tests are conducted. The UAM traffic density is taken from 10 to 300 (flights/hour/36km^2) with an interval of ten. To reduce the random effects of individual flight missions, flights in the next traffic density scenarios are obtained by adding all current flights with ten new flights. Delayed flights are randomly chosen in each density scenario with a percentage of 20, which is considered the threshold of the complete disruption of initial optimized plans (Fig. 7.6(a)). Lastly, the computation time of the HSS algorithm in each scenario is recorded. All obtained results are shown in Fig. 7.8.

The 20% delayed flights in increasing traffic density scenarios cause more disruptions to the initially optimized flight trajectories, reflected by a notable upward trend in the number of conflicts (blue curve in Fig. 7.8(a)). After optimization, all conflicts are solved (red curve in Fig. 7.8(a)), however, the traffic system pays a cost of an increased number of flight rejections (Fig. 7.8(b)) with the increase of the traffic density. Within a density of 150, conflicts can be solved through speed adjustment with only 5 flights being rejected due to their departure time exceeding the departure window. However, the average delays doubled from 25 seconds to 50 seconds per flight compared with the initial plans before optimization (Fig. 7.8(c)). That is because the average flight speed is reduced in high-density scenarios (Fig. 7.8(d)) to avoid conflicts, increasing the total airborne delay. With the increase of the traffic density, the average flight speed first saw a slight upward trend from 8 m/s to 11 m/s at the density of 110, while a downward trend dominates with the average speed drops to 9 m/s when reaching a density of 290 (flights/hour/36km^2).

Lastly, the computation time of the HSS algorithm for each density scenario is presented in Fig. 7.8(e). With the increase in traffic density, the computation time shows a low-slope linear trend, which demonstrates our analysis of the computation complexity $O(n)$ in Section IV-D. Below the density of 100, the proposed HSS algorithm can solve the UTFM optimization problem within 5 seconds, and 12 seconds even for a density of 300, which is equivalent to 6,000 flights/hour in Singapore urban airspace below 400 ft. That makes the HSS algorithm a practical solution to handle high-density UAM traffic flow optimization problems.

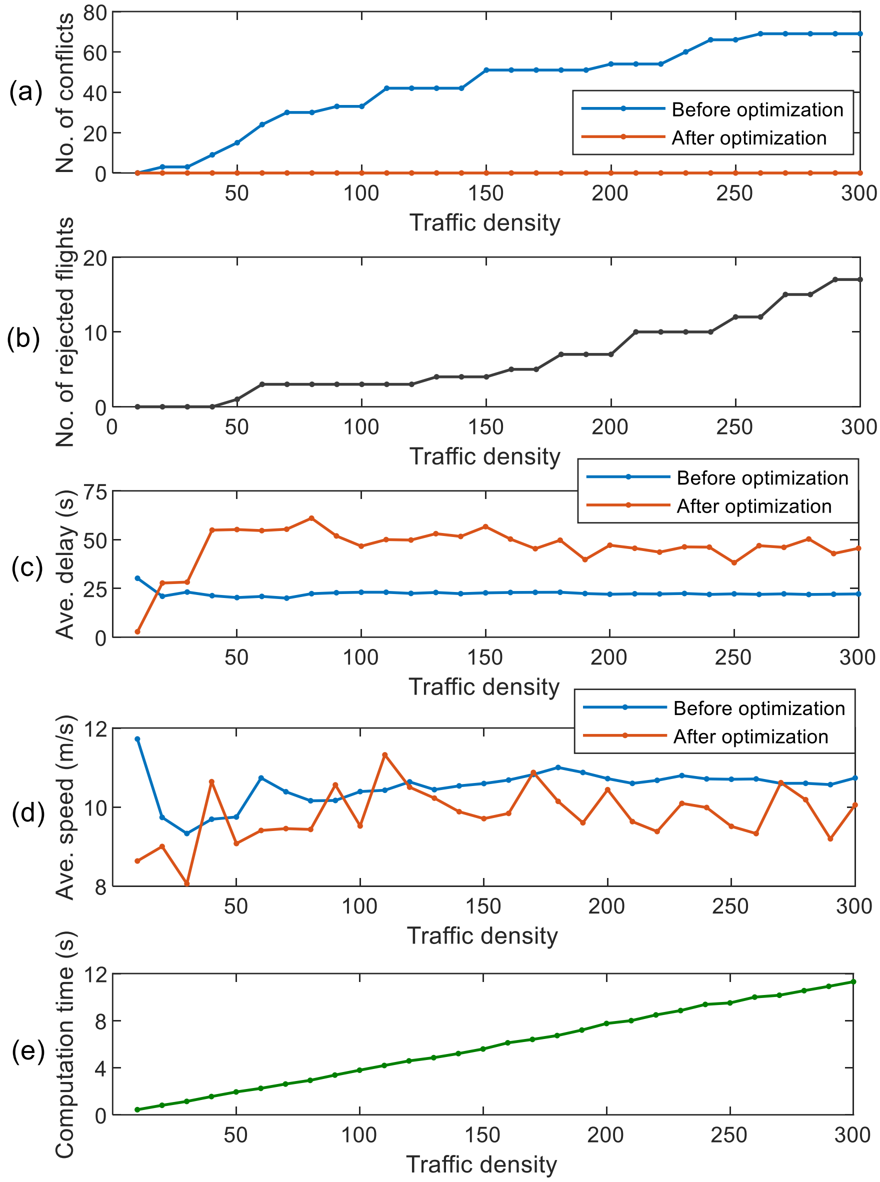


Fig. 7.8: Scalability analysis of the optimization model and the HSS algorithm in scenarios with different traffic densities. (flights/hour/36km^2)

7.6 Summary

In this work, a chance-constrained urban air mobility (UAM) traffic flow optimization model is proposed to optimize the disrupted four-dimensional trajectories due to flight delays and trajectory uncertainties with the purposes of 1) ensuring system safety by resolving all conflicts caused by the uncertainties, and 2) improving system resilience by recovering delayed flights with robust and conflict-free trajectories. To solve the optimization model, a hierarchical stochastic search algorithm is developed and validated in various traffic patterns and densities. Lastly, simulations are performed to demonstrate the effectiveness, reliability, and scalability of the proposed optimization model and HSS algorithm. From the obtained results, we demonstrate that:

(i) The chance-constrained optimization model performs well in terms of solving flight conflicts in the presence of uncertainties. The model can successfully solve all conflicts in a wide range of traffic density scenarios only at little cost to the system efficiency, e.g., increased number of flight rejections and average flight delays.

(ii) The chance-constrained model enables the connection between the user's risk preferences and the choice of the trajectory uncertainty distribution (shown in Table 7.2). Given a risk level α, a required safe separation will be derived from the uncertainty distribution of trajectory deviations, which has been tested as a normal distribution (at a 95% confidence level) through the flight experiment data.

(iii) The proposed hierarchical stochastic search algorithm can solve the optimization model with a safety guarantee of resolving all conflicts. This is achieved by increasing the weights for conflict penalty through the weight update function, as shown in Algorithm 1. The computation of the hierarchical stochastic search algorithm can be completed in polynomial time, which makes it a practical solution algorithm to solve high-density UAM traffic flow optimization problems in various environments.

(iv) Sensitivity analysis results show that the optimization performance is sensitive to traffic density but has no significant changes to traffic patterns in different environments. Given the same environment, the airspace capacity drops with the decrease of the risk level. Meaning that a smaller risk level (i.e., higher safety requirement) will have a cost to the capacity of the UAM traffic system.

In summary, our proposed chance-constrained UTFM optimization model can significantly improve the safety and resilience of the UAM traffic system under flight delay and trajectory uncertainties with little cost to the system efficiency (e.g., delays). Which forms another safety assurance layer for the UAM traffic system, filling the gap between the strategic and tactical safety assurance layers.

Chapter 8

8 Conclusion and Future Work

In this Chapter, we summarize the research works conducted on risk assessment and mitigation for safe UAS operations. Followed by discussions of some future research directions based on our current works.

8.1 Major Research Findings

Third-party risk assessment and mitigation is one of the key challenges for the introduction of unmanned aircraft system operations into urban low-altitude airspace. For one thing, drone operations may fail and crash into people causing ground risk. For another, airborne collisions between drone yield air risk. These safety risks need to be quantitatively modeled and mitigated to enable safe and resilient UAS operations. This thesis, therefore, focuses on developing a Third-party Risk Assessment and Mitigation System (TRAMS) by leveraging mathematical optimization and machine learning techniques. TRAMS considers both ground risk and air risk with the objective of minimizing safety risks and improving the efficiency and resilience of UAS operations.

First, we introduce a risk-based AirMatrix concept of operations (ConOps), which enables digitalized low-altitude urban airspace management by incorporating various data sources, such as third-party risk, flight plans, communication, navigation, and surveillance of unmanned aircraft. The risk-based AirMatrix ConOps provides both safety assurance and flight flexibility for unmanned aircraft system (UAS) traffic management. (Achieved in Chapter 3)

Second, we study the UAV flight path optimization problem with consideration of an integrated cost assessment model of fatality risk, property damage risk, and noise impact.The developed flight path optimization method is effective and reliable in minimizing the risk cost of the flight path. Besides, our developed cost-based path planning algorithm provides a novel scheme to determine the heuristic information for path searching methods in cost-based environments. The simulation results show the effectiveness and reliability of the developed algorithm in solving cost-based path optimization problems. This study allows the integrated assessment for various third-party risk factors to generate a unified risk-cost map. With the map, the safety risk of an individual flight can be reduced by the path optimization model. (Achieved in Chapter 4)

Third, the risk assessment results from the above work can be influenced by risk uncertainties due to population movement. Hence, we investigate the stochastic route optimization problem under dynamic ground risk for safe and robust UAS operations. The problem is developed as a two-stage stochastic route optimization model. The first-stage decision handles the route feasibility requirements regarding the target level of safety and flight duration. While the route selection and departure time-related decisions are made in the second stage to cope with stochastic risk conditions. Simulation results demonstrate the good performance of the proposed model in terms of ground risk mitigation. The research findings of this work can provide support for (i) aviation authorities to decide if a specific flight plan (with flight route and estimated time of departure) can be approved under a given target level of safety, (ii) eVTOL users (delivery or air taxi companies) to optimize the routes and timetables of their flight missions to avoid high-risk conditions. Which ultimately contributes to the safety risk management of UAS operations in low-altitude urban airspace. (Achieved in Chapter 5)

Fourth, mid-air collision between UAVs is considered as air risk, and the collided UAVs will fall and cause knock-on ground risk. To reduce these risks, we study flight conflict detection and resolution problems for large-scale high-density UAS traffic flow. An adaptive conflict resolution framework is proposed to optimize the strategies used for different types of conflicts with explainable mechanisms, and with the goal of optimizing UAV 4D routes. The framework is formulated as a double-layer optimization problem, and scheduling, speed adjustment, and rerouting strategies are considered for conflict resolution. The first layer of the framework is established as a probabilistic selection model to make decisions on which strategy is selected for what type of conflict. The second layer is developed as a mixed integer nonlinear programming (MINLP) model to optimize the decision variables of the strategies selected by the first layer. A novel meta-heuristic stochastic fractal search (SFS) algorithm is then developed to solve the proposed problems. Simulation results demonstrate the effectiveness and robustness of the proposed framework and algorithms. (Achieved in Chapter 6)

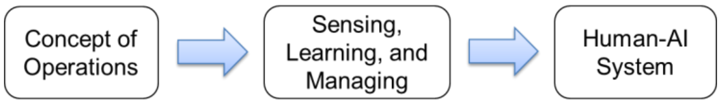
Lastly, we study the robust unmanned air traffic flow management (UTFM) problems under uncertainties of flight delay and trajectory deviations, which contribute to safe and resilient UAS traffic flow management. A chance-constrained urban air mobility traffic flow optimization model is proposed to optimize the disrupted four-dimensional trajectories due to flight delays and trajectory uncertainties with the purposes of (i) ensuring system safety by resolving all conflicts caused by the uncertainties, and (ii) improving system resilience by recovering delayed flights with robust and conflict-free trajectories. To solve the optimization model, a hierarchical stochastic search algorithm is developed and validated in various traffic patterns and densities. Lastly, simulations are performed to demonstrate the effectiveness, reliability, and scalability of the proposed optimization model and hierarchical stochastic search algorithm. (Achieved in Chapter 7)

In summary, the research works done in this thesis is to form a third-party risk assessment and mitigation system (TRAMS) by combining ground risk studies (Chapter 3, Chapter 4, and Chapter 5) and air risk studies (Chapter 6 and Chapter 7). The ground risk studies produce a risk-minimizing route for individual flight missions. Based on the generated risk-minimizing route, the air risk studies handle multiple flight missions to revolve flight conflicts and disruptions under uncertainties. Incorporated ground risk and air risk studies for third-party people, the TRAMS enables safe, resilient, and efficient UAS operations with flight recovery capability under uncertainties in urban environments.

This thesis presents some research works on safety risk management and traffic flow management for emerging unmanned aircraft system operations, and it can be further extended in several ways.

8.2 Future Research Directions

Unmanned aircraft system operations, as an emerging domain, have been evolving fast with novel concept of operations such as advanced air mobility (AAM), which employes various types of eVTOLs to expand the boundaries of air transport services into urban, suburban, and rural areas. The mixed types of eVTOLs and various missions bring new challenges for risk identification, assessment, and mitigation, which require considerable research efforts. While emerging data sources such as demand distribution data and flight accident dataset could enable better sensing, learning, and managing capabilities for more accurate and comprehensive analysis of UAS risks. Moreover, research works on the development of hybrid human-AI system would also be needed to enable mixed high-density UAS traffic flow operating in complex low-altitude urban airspace. Specifically, this research work can be extended from the following aspects.



(i). The concepts of operations for drone operations are still an open question, which needs considerable efforts to develop for areas such as air traffic service for UAS operations, airspace management for low-altitude urban airspace, and unmanned air traffic flow management concepts.

(ii). Collision risk assessment between UAS and manned aircraft. The environment of this thesis is in low-altitude urban areas, and the integrated cost assessment model does not consider the risk cost of collision between UAVs with general aviation and commercial aircraft. The risk assessment models for air traffic conflict and collision [31] for a mixed type of aircraft require further developments.

(iii). Risk modeling for advanced air mobility (AAM) with heterogeneous UAVs. This thesis considers the homogenous unmanned aircraft (UA) type for risk assessment and risk-based route optimization. The results can be different for heterogeneous UAs due to the difference in eVTOL size, weight, reliability (or failure rate), and operational purpose (parcel delivery or passenger transport) [270]. The risk modeling and assessment for various types of eVTOLs are needed to enable safe and harmonious operations for advanced air mobility [271].

(iv). Risk mitigation approaches. In this thesis, risk mitigation is conducted from a perspective of air traffic management. The key is to perform flight route optimization to avoid high-risk areas with high population density. Other mitigation approaches can also be investigated. For instance, from a design perspective, eVTOLs can be equipped with parachutes to reduce the impact speed, which subsequently reduces the impact fatality probability. Besides, educating people to protect their head on response to a UAV crash accident also helps. As current crash severity assessment models assume the direct impact to a dummy head.

(v). Tactical conflict detection and resolution in robotic levels. This thesis studies conflict resolution at strategic and pre-tactical levels to generate robust and conflict-free 4D flight trajectories. In the next steps, we can investigate tactical decision-making approaches for real-time conflict resolution to handle dynamic risks. There are also some other methods such as game theory [272] and deep reinforcement learning [273] have shed light on solving real-time conflict resolution problems, much more effects are still required to make these methods robust and implementable for real-world UAV operations.

(vi). Integrated air traffic flow management for mixed types of UAVs. In the advanced air mobility concept of operations [137], there will be a mixed type of eVTOLs operating in urban airspace. Follow-up works are therefore needed to develop an unmanned air traffic flow optimization model considering various types of eVTOLs that have different specifications, flight performances, and trajectory uncertainties. In the models, apart from the trajectory uncertainty, other types of uncertainties could also impact the UAM traffic flow, such as probabilistic Geo-Fencing [259]. Future works are required to incorporate these uncertainties into the proposed UTFM model to ensure no violations of Geo-Fencings (e.g., restricted areas) and to mitigate dynamic ground risks to third parties in urban environments.

Appendix: List of Publications

Journal Publications

[1] **Bizhao Pang**, Xinting Hu, and Kin Huat Low. Stochastic Route Optimization under Dynamic Ground Risk Uncertainties for Safe Drone-Based Delivery Operations. 2023, Under review.

[2] **Bizhao Pang** and Kin Huat Low. Chance-Constrained UAM Traffic Flow Optimization with Fast Disruption Recovery Under Delay and Trajectory Uncertainty. 2023, Under review.

[3] **Bizhao Pang**, Kin Huat Low, and Chen Lv. Adaptive Conflict Resolution for Multi-UAV 4D Routes Optimization Using Stochastic Fractal Search Algorithm. *Transportation Research Part C: Emerging technology*, 2022. Doi: 10.1016/j.trc.2022.103666

[4] **Bizhao Pang**, Xinting Hu, Wei Dai, and Kin Huat Low. UAV path optimization with an integrated cost assessment model considering third-party risks in metropolitan environments. *Reliability Engineering & System Safety*, 2022, 108399. Doi: 10.1016/j.ress.2022.108399

[5] **Bizhao Pang**, Wei Dai, Xinting Hu, Fuqing Dai, and Kin Huat Low. Multiple air route crossing waypoints optimization via artificial potential field method. *Chinese Journal of Aeronautics.* 34, no. 4 (2021): 279-292. Doi: 10.1016/j.cja.2020.10.008

[6] Wei Dai, **Bizhao Pang**, and Kin Huat Low. "Conflict-free four-dimensional path planning for urban air mobility considering airspace occupancy." *Aerospace Science and Technology* 119 (2021): 107154. Doi: 10.1016/j.ast.2021.107154

[7] Yu Wu, Kin Huat Low, **Bizhao Pang**, and Qingyu Tan. Swarm-based 4D Path Planning for Drone Operations in Urban Environments. *IEEE Transactions on Vehicular Technology,* (2021), 70(8), 7464-7479. Doi: 10.1109/TVT.2021.3093318

[8] Hu, Xinting, **Bizhao Pang**, Fuqing Dai, and Kin Huat Low. "Risk Assessment Model for UAV Cost-Effective Path Planning in Urban Environments." *IEEE Access* 8 (2020): 150162-150173. Doi: 10.1109/ACCESS.2020.3016118

Conference Publications

[1] **Bizhao Pang**, Hu, Xinting, Yi Yang Poh, and Kin Huat Low. Population Density

Estimation for Dynamic Ground Risk Assessment of Drone Operations. *In 2023 IEEE/AIAA 42nd Digital Avionics Systems Conference (DASC)*. (Accepted)

[2] **Bizhao Pang**, Mingcheng Zhang, Chao Deng, and Kin Huat Low. "Investigation of Flight Technical Error for UAV Separation Requirement Based on Flight Trajectory Data." *In AIAA Aviation Forum*, p. 3763. 2022.

[3] **Bizhao Pang**, C.H. John Wang, Kin Huat Low. Framework of Level-of-Autonomy-based Concept of Operations: UAS Capabilities. *In 2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC) (pp. 1-10). IEEE*.

[4] **Bizhao Pang**, Wei Dai, Thu Ra, Kin Huat Low. "A Concept of Airspace Configuration and Operational Rules for UAS in Current Airspace." In *2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC)*, pp. 1-9. IEEE, 2020.

[5] **Bizhao Pang**, Ng Ee Meng, Kin Huat Low. UAV Trajectory Estimation and Deviation Analysis for Contingency Management in Urban Environments. 2020 *AIAA Aviation Forum*, Nevada, USA, p. 2919.

[6] **Bizhao Pang,** Qingyu Tan, Kin Huat Low. A Risk Model Based Traffic Network for Adaptive Urban Airspace Management. 2020 *AIAA Aviation Forum*, Nevada, USA, p. 2900.

[7] **Bizhao Pang**, Ng Ee Meng, Kin Huat Low. Contingency Management of Urban UTM (UAS Traffic Management) Based on Airspace Contexts. *International Civil Aviation Organization (ICAO) Unmanned Aircraft Systems (UAS) Industry Symposium*, Montreal, Canada, 2019.

[8] Lee, Lewis, **Bizhao Pang**, and Kin Huat Low. "Environmental Data Analytics for Safe Drone Operations in Low-Altitude Urban Environments." *In AIAA Aviation Forum*, p. 3405. 2022.

[9] Hu, Xinting, **Bizhao Pang**, and Kin Huat Low. "Wind patterns analysis on temporal scales for safe UAV operations using statistical approaches." 2022 *IEEE/AIAA 41st Digital Avionics Systems Conference (DASC). IEEE*, 2022.

[10] Wei Dai, **Bizhao Pang**, Kin Huat Low. Accessibility Analysis of Unmanned Aerial Vehicles Near Airports with a Four-Dimensional Airspace Management Concept. *IEEE/AIAA 39th Digital Avionics Systems Conference (DASC 2020)*, 11-15 Oct 2020, San Antonio, TX, United States.

Bibliography

[1] EUROCONTROL, “U-space Concept of Operations,” *SESAR Joint Undertaking*, vol. 1, no. October 2019, pp. 1–92, 2019.

[2] FAA and NASA, “Urban Air mobility (UAM) Concept of Operations v1.0,” 2020. [Online]. Available: https://utm.arc.nasa.gov/docs/2018-UTM-ConOps-v1.0.pdf

[3] T. Kirschstein, “Comparison of energy demands of drone-based and ground-based parcel delivery services,” *Transp Res D Transp Environ*, vol. 78, no. December 2019, p. 102209, 2020, doi: 10.1016/j.trd.2019.102209.

[4] M. Doole, J. Ellerbroek, and J. Hoekstra, “Estimation of traffic density from drone-based delivery in very low-level urban airspace,” *J Air Transp Manag*, vol. 88, Sep. 2020, doi: 10.1016/j.jairtraman.2020.101862.

[5] K. Balakrishnan, J. Polastre, J. M. R. Golding, and P. Sachs, “Blueprint for the Sky - The Roadmap for the Safe Integration of Autonomous Aircraft,” 2018.

[6] X. Edward, “White paper on UAM system,” 2020. [Online]. Available: https://www.ehang.com/app/en/EHang White Paper on Urban Air Mobility Systems.pdf

[7] F. Saeed, A. Mehmood, M.F. Majeed, C. Maple, K. Saeed, M.K. Khattak, H. Wang and G. Epiphaniou, “Smart delivery and retrieval of swab collection kit for COVID-19 test using autonomous Unmanned Aerial Vehicles,” *Physical Communication*, vol. 48, p. 101373, 2021, doi: 10.1016/j.phycom.2021.101373.

[8] R. Melnyk, D. Schrage, V. Volovoi, and H. Jimenez, “A third-party casualty risk model for unmanned aircraft system operations,” *Reliab Eng Syst Saf*, vol. 124, pp. 105–116, 2014, doi: 10.1016/j.ress.2013.11.016.

[9] H. A. P. Blom, C. Jiang, W. B. A. Grimme, M. Mitici, and Y. S. Cheung, “Third party risk modelling of Unmanned Aircraft System operations, with application to parcel delivery service,” *Reliab Eng Syst Saf*, vol. 214, no. April, p. 107788, 2021, doi: 10.1016/j.ress.2021.107788.

[10] R. A. Clothier, B. P. Williams, and K. J. Hayhurst, “Modelling the risks remotely piloted aircraft pose to people on the ground,” *Saf Sci*, vol. 101, no. August, pp. 33–47, 2018, doi: 10.1016/j.ssci.2017.08.008.

[11] C. Jiang, H. A. P. Blom, and A. Sharpanskykh, “Third Party Risk Indicators and Their Use in Safety Regulations for UAS Operations,” in *AIAA AVIATION 2020 FORUM, 2020*, AIAA, 2020, pp. 1–15. doi: 10.2514/6.2020-2901.

[12] L. K. Lin Tan, B. C. Lim, G. Park, K. H. Low, and V. C. Seng Yeo, “Public acceptance of drone applications in a highly urbanized environment,” *Technol Soc*, vol. 64, no. November 2020, p. 101462, 2021, doi: 10.1016/j.techsoc.2020.101462.

[13] JARUS, “JARUS guidelines on Specific Operations Risk Assessment (SORA),” 2017.

[14] R. A. Clothier, J. L. Palmer, R. A. Walker, and N. L. Fulton, “Definition of an airworthiness certification framework for civil unmanned aircraft systems,” *Saf Sci*, vol. 49, no. 6, pp. 871–885, 2011, doi: 10.1016/j.ssci.2011.02.004.

[15] P. Gonçalves, J. Sobral, and L. A. Ferreira, “Unmanned aerial vehicle safety assessment modelling through petri Nets,” *Reliab Eng Syst Saf*, vol. 167, no. June, pp. 383–393, 2017, doi: 10.1016/j.ress.2017.06.021.

[16] L.C. Barr, R. Newman, E. Ancel, C.M. Belcastro, J.V. Foster, J. Evans, D.H. Klyde, “Preliminary Risk Assessment for Small Unmanned Aircraft Systems,” in *17th AIAA Aviation Technology, Integration, and Operations Conference*, 2017. doi: 10.2514/6.2017-3272.

[17] A. Allouch, A. Koubaa, M. Khalgui, and T. Abbes, “Qualitative and Quantitative Risk Analysis and Safety Assessment of Unmanned Aerial Vehicles Missions over the Internet,” *IEEE Access*, vol. 7, pp. 53392–53410, 2019, doi: 10.1109/ACCESS.2019. 2911980.

[18] N. Pongsakornsathien, S. Bijjahalli, A. Gardi, A. Symons, Y. Xi, R. Sabatini, T. Kistan, “A performance-based airspace model for unmanned aircraft systems traffic management,” *Aerospace*, vol. 7, no. 11, pp. 1–25, 2020, doi: 10.3390/aerospace 7110154.

[19] B. Pang, E. M. Ng, and K. H. Low, “UAV Trajectory Estimation and Deviation Analysis for Contingency Management in Urban Environments,” in *AIAA AVIATION 2020 FORUM*, AIAA, 2020, pp. 1–10. doi: 10.2514/6.2020-2919.

[20] J. Kirtner and H. Anderson, “The Application of Land Use/Land Cover (Clutter) Data to Wireless Communication System Design,” 2008.

[21] H. Usui, "Statistical distribution of building lot depth: Theoretical and empirical investigation of downtown districts in Tokyo," *Environ Plan B Urban Anal City Sci*, vol. 46, no. 8, pp. 1499–1516, 2019, doi: 10.1177/2399808319840366.

[22] X. Hu, B. Pang, F. Dai, and K. H. Low, "Risk Assessment Model for UAV Cost-Effective Path Planning in Urban Environments," *IEEE Access*, vol. 8, pp. 150162–150173, 2020, doi: 10.1109/ACCESS.2020.3016118.

[23] K. Dalamagkidis, K. P. Valavanis, and L. A. Piegl, "Evaluating the risk of unmanned aircraft ground impacts," in *16th Mediterranean Conference on Control and Automation*, 2008, pp. 709–716. doi: 10.1109/MED.2008.4602249.

[24] P. D. Vascik and R. J. Hansman, "Scaling constraints for urban air mobility operations: Air traffic control, ground infrastructure, and noise," *2018 Aviation Technology, Integration, and Operations Conference*, pp. 1–25, 2018, doi: 10.2514/6.2018-3849.

[25] B. Pang, Q. Tan, T. Ra, and K. H. Low, "A risk-based uas traffic network model for adaptive urban airspace management," in *Aiaa Aviation 2020 Forum*, 2020, pp. 1–9. doi: 10.2514/6.2020-2900.

[26] M. Ghasri and M. Maghrebi, "Factors affecting unmanned aerial vehicles' safety: A post-occurrence exploratory data analysis of drones' accidents and incidents in Australia," *Saf Sci*, vol. 139, p. 105273, 2021, doi: 10.1016/j.ssci.2021.105273.

[27] JARUS, "JARUS guidelines on Specific Operations Risk Assessment (SORA), EDITION 2.0," 2019. [Online]. Available: http://jarus-uas.org

[28] ICAO, "Icao Doc 9859 (Safety Management System)," 2017.

[29] FAA, "Operation of small, unmanned aircraft systems over people," 2021.

[30] C. H. Koh, K.H. Low, L. Li, Y. Zhao, C. Deng, S.K. Tan, Y. Chen, B.C Yeap. and X Li, "Weight threshold estimation of falling UAVs (Unmanned Aerial Vehicles) based on impact energy," *Transp Res Part C Emerg Technol*, vol. 93, no. April, pp. 228–255, 2018, doi: 10.1016/j.trc.2018.04.021.

[31] M. Mitici and H. A. P. Blom, "Mathematical Models for Air Traffic Conflict and Collision Probability Estimation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 3, pp. 1052–1068, 2019, doi: 10.1109/TITS.2018.2839344.

[32] S. Bertrand, N. Raballand, and F. Viguier, "Evaluating Ground Risk for Road Networks Induced by UAV Operations," in *2018 International Conference on Unmanned Aircraft Systems, ICUAS 2018*, IEEE, 2018, pp. 168–176. doi: 10.1109/ICUAS.2018.8453441.

[33] J. Breunig, J. Forman, S. Sayed, L. Audenaerd, A. Branch, and M. Hadjimichael, “Modeling risk-based approach for small, unmanned aircraft systems,” in *AUVSI XPONENTIAL 2019: All Things Unmanned*, 2019, pp. 1–23.

[34] B. Levasseur, S. Bertrand, N. Raballand, F. Viguier, and G. Goussu, “Accurate Ground Impact Footprints and Probabilistic Maps for Risk Analysis of UAV Missions,” in *IEEE Aerospace Conference Proceedings*, 2019, pp. 1–10. doi: 10.1109/AERO.2019. 8741718.

[35] S. H. Kim, “Third-party risk analysis of small, unmanned aircraft systems operations,” *Journal of Aerospace Information Systems*, vol. 17, no. 1, pp. 24–35, 2020, doi: 10.2514/ 1.I010763.

[36] X. Ren and C. Cheng, “Model of Third-Party Risk Index for Unmanned Aerial Vehicle Delivery in Urban Environment,” *Sustainability*, vol. 12, no. 20, p. 8318, 2020, doi: 10.3390/su12208318.

[37] J. Votion and Y. Cao, “Diversity-based cooperative multivehicle path planning for risk management in costmap environments,” *IEEE Transactions on Industrial Electronics*, vol. 66, no. 8, pp. 6117–6127, 2019, doi: 10.1109/TIE.2018.2874587.

[38] A. la Cour-Harbo and H. Schiøler, “Probability of Low-Altitude Midair Collision Between General Aviation and Unmanned Aircraft,” *Risk Analysis*, vol. 39, no. 11, pp. 2499–2513, 2019, doi: 10.1111/risa.13368.

[39] S. Chaimatanan, D. Delahaye, and M. Mongeau, “A hybrid metaheuristic optimization algorithm for strategic planning of 4D aircraft trajectories at the continental scale,” *IEEE Comput Intell Mag*, vol. 9, no. 4, pp. 46–61, 2014, doi: 10.1109/MCI.2014.2350951.

[40] F. H. C. Dias and D. Rey, “Robust aircraft conflict resolution under trajectory prediction uncertainty,” *ArXiv*, 2020.

[41] F. H. C. Dias, H. Hijazi, and D. Rey, “Disjunctive linear separation conditions and mixed-integer formulations for aircraft conflict resolution,” *Eur J Oper Res*, vol. 296, no. 2, pp. 520–538, 2022, doi: 10.1016/j.ejor.2021.03.059.

[42] D. González-Arribas, M. Soler, and M. Sanjurjo-Rivo, “Robust aircraft trajectory planning under wind uncertainty using optimal control,” *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 3, pp. 673–688, 2018, doi: 10.2514/1. G002928.

[43] W. X. Lim and Z. W. Zhong, "Re-Planning of Flight Routes Avoiding Convective Weather and the 'Three Areas,'" *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 868–877, 2018, doi: 10.1109/TITS.2017.2705098.

[44] H. Xie, K. H. Low, and Z. He, "Adaptive visual servoing of unmanned aerial vehicles in GPS-denied environments," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 6, pp. 2554–2563, 2017, doi: 10.1109/TMECH.2017.2755669.

[45] H. Liu, M. H. Che Man, and K. H. Low, "UAV airborne collision to manned aircraft engine: Damage of fan blades and resultant thrust loss," *Aerosp Sci Technol*, vol. 113, p. 106645, 2021, doi: 10.1016/j.ast.2021.106645.

[46] C. H. J. Wang, S. K. Tan, and K. H. Low, "Collision risk management for non-cooperative UAS traffic in airport-restricted airspace with alert zones based on probabilistic conflict map," *Transp Res Part C Emerg Technol*, vol. 109, no. July, pp. 19–39, 2019, doi: 10.1016/j.trc.2019.09.017.

[47] C. H. J. Wang, S. K. Tan, and K. H. Low, "Three-dimensional (3D) Monte-Carlo modeling for UAS collision risk management in restricted airport airspace," *Aerosp Sci Technol*, vol. 105, p. 105964, 2020, doi: 10.1016/j.ast.2020.105964.

[48] P. D. Vascik and R. J. Hansman, "Assessing integration between emerging and conventional operations in urban airspace," in *AIAA Aviation 2019 Forum*, 2019, pp. 1–24. doi: 10.2514/6.2019-3125.

[49] W. Dai, B. Pang, and K. H. Low, "Accessibility Analysis of Unmanned Aerial Vehicles Near Airports with a Four-Dimensional Airspace Management Concept," in *2020 IEEE/AIAA 39th Digital Avionics Systems Conference (DASC)*, IEEE, 2020, pp. 1–9.

[50] N. Doan, V. Duong, and D. Nace, "The Air-Route Network Design Problem," in *Intl. Conf. RIVF*, 2004, pp. 15–20.

[51] X. He, F. He, L. Li, L. Zhang, and G. Xiao, "A route network planning method for urban air delivery," *Transp Res E Logist Transp Rev*, vol. 166, Oct. 2022, doi: 10.1016/j.tre. 2022.102872.

[52] B. Pang, W. Dai, X. Hu, F. Dai, and K. H. Low, "Multiple air route crossing waypoints optimization via artificial potential field method," *Chinese Journal of Aeronautics*, vol. 34, no. 4, 2021, doi: 10.1016/j.cja.2020.10.008.

[53] J. Cho and Y. Yoon, “How to assess the capacity of urban airspace: A topological approach using keep-in and keep-out geofence,” *Transp Res Part C Emerg Technol*, vol. 92, no. May, pp. 137–149, 2018, doi: 10.1016/j.trc.2018.05.001.

[54] J. M. Hoekstra, J. Maas, M. Tra, and S. E., “How Do Layered Airspace Design Parameters Affect Airspace Capacity and Safety?” no. June, 2016, Accessed: Apr. 16, 2022. [Online]. Available: https://pure.tudelft.nl/ws/portalfiles/portal/87262684/ICRAT_2016_paper_6.pdf

[55] B. Pang, W. Dai, T. Ra, and K. H. Low, “A Concept of Airspace Configuration and Operational Rules for UAS in Current Airspace,” in *2020 IEEE/AIAA 39th Digital Avionics Systems Conference (DASC)*, IEEE, 2020, pp. 1–9. doi: 10.1109/DASC50938.2020.9256627.

[56] P. Pierpaoli and A. Rahmani, “UAV collision avoidance exploitation for noncooperative trajectory modification,” *Aerosp Sci Technol*, vol. 73, pp. 173–183, 2018, doi: 10.1016/j.ast.2017.12.008.

[57] P. Hang, C. Lv, Y. Xing, C. Huang, and Z. Hu, “Human-Like Decision Making for Autonomous Driving: A Noncooperative Game Theoretic Approach,” *IEEE Transactions on Intelligent Transportation Systems*, no. 1922500046, pp. 1–10, 2020, [Online]. Available: http://arxiv.org/abs/2005.11064

[58] B. Aydin, “Public acceptance of drones: Knowledge, attitudes, and practice,” *Technol Soc*, vol. 59, no. July, p. 101180, 2019, doi: 10.1016/j.techsoc.2019.101180.

[59] A. J. Torija, Z. Li, and R. H. Self, “Effects of a hovering unmanned aerial vehicle on urban soundscapes perception,” *Transp Res D Transp Environ*, vol. 78, p. 102195, 2020, doi: 10.1016/j.trd.2019.11.024.

[60] X. Zhang, H. Bian, R. Fattah, and Y. Sun, “Acoustic mapping of drones in complex environments,” in *Proceedings of the 26th International Congress on Sound and Vibration, ICSV 2019*, 2019, pp. 1–8.

[61] M. W. Bauer, “First assessment of community noise for a simulated scenario of new urban air traffic,” in *Proceedings of the 26th International Congress on Sound and Vibration, ICSV 2019*, 2019, pp. 1–8.

[62] X. He, C. Jiang, L. Li, and H. Blom, “A Simulation Study of Risk-Aware Path Planning in Mitigating the Third-Party Risk of a Commercial UAS Operation in an Urban Area,” *Aerospace*, vol. 9, no. 11, Nov. 2022, doi: 10.3390/aerospace9110682.

[63] H. Ergezer and K. Leblebicioglu, “Path planning for UAVs for maximum information collection,” *IEEE Trans Aerosp Electron Syst*, vol. 49, no. 1, pp. 502–520, 2013, doi: 10.1109/TAES.2013.6404117.

[64] H. Cicibas, K. A. Demir, and N. Arica, “Comparison of 3D versus 4D path planning for unmanned aerial vehicles,” *Def Sci J*, vol. 66, no. 6, pp. 651–664, 2016, doi: 10.14429/dsj.66.9575.

[65] X. Liu, X. Du, X. Zhang, Q. Zhu, and M. Guizani, “Evolution-algorithm-based unmanned aerial vehicles path planning in complex environment,” *Computers and Electrical Engineering*, vol. 80, p. 106493, 2019, doi: 10.1016/j.compeleceng.2019.106493.

[66] C. C. Murray and W. Park, “Incorporating human factor considerations in unmanned aerial vehicle routing,” *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, vol. 43, no. 4, pp. 860–874, 2013, doi: 10.1109/TSMCA.2012.2216871.

[67] P. Yang, K. Tang, J. A. Lozano, and X. Cao, “Path Planning for Single Unmanned Aerial Vehicle by Separately Evolving Waypoints,” *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1130–1146, 2015, doi: 10.1109/TRO.2015.2459812.

[68] F. Furini, C. A. Persiani, and P. Toth, “The Time Dependent Traveling Salesman Planning Problem in Controlled Airspace,” *Transportation Research Part B: Methodological*, vol. 90, pp. 38–55, 2016, doi: 10.1016/j.trb.2016.04.009.

[69] C. Yin, Z. Xiao, X. Cao, X. Xi, P. Yang, and D. Wu, “Offline and Online Search: UAV Multiobjective Path Planning Under Dynamic Urban Environment,” *IEEE Internet Things J*, vol. 5, no. 2, pp. 546–558, 2018, doi: 10.1109/JIOT.2017.2717078.

[70] D. P. Bertsekas, “Robust shortest path planning and semicontractive dynamic programming,” *Naval Research Logistics*, vol. 66, no. 1, pp. 15–37, 2019, doi: 10.1002/nav.21697.

[71] D. P. Bertsekas, “Affine Monotonic and Risk-Sensitive Models in Dynamic Programming,” *IEEE Trans Automat Contr*, vol. 64, no. 8, pp. 3117–3128, 2019, doi: 10.1109/TAC.2019.2896049.

[72] H. Huang, A. V. Savkin, and C. Huang, “Reliable Path Planning for Drone Delivery Using a Stochastic Time-Dependent Public Transportation Network,” *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, 2020, doi: 10.1109/TITS.2020.2983491.

[73] N. Wen, L. Zhao, X. Su, and P. Ma, "UAV online path planning algorithm in a low altitude dangerous environment," *IEEE/CAA Journal of Automatica Sinica*, vol. 2, no. 2, pp. 173–185, 2015, doi: 10.1109/JAS.2015.7081657.

[74] H. Oleynikova, M. Burri, Z. Taylor, J. Nieto, R. Siegwart, and E. Galceran, "Continuous-time trajectory optimization for online UAV replanning," *IEEE International Conference on Intelligent Robots and Systems*, vol. 2016-Novem, pp. 5332–5339, 2016, doi: 10.1109/IROS.2016.7759784.

[75] Y. Lin and S. Saripalli, "Sampling-Based Path Planning for UAV Collision Avoidance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 11, pp. 3179–3192, 2017, doi: 10.1109/TITS.2017.2673778.

[76] Z. Zhang, R. Du, and R. V. Cowlagi, "Randomized sampling-based trajectory optimization for UAVs to satisfy linear temporal logic specifications," *Aerosp Sci Technol*, vol. 96, p. 105591, 2020, doi: 10.1016/j.ast.2019.105591.

[77] J. D. Marble and K. E. Bekris, "Asymptotically near-optimal planning with probabilistic roadmap spanners," *IEEE Transactions on Robotics*, vol. 29, no. 2, pp. 432–444, 2013, doi: 10.1109/TRO.2012.2234312.

[78] A. C. Woods and H. M. La, "A Novel Potential Field Controller for Use on Aerial Robots," *IEEE Trans Syst Man Cybern Syst*, vol. 49, no. 4, pp. 665–676, 2019, doi: 10.1109/TSMC.2017.2702701.

[79] J. P. Wilhelm and G. Clem, "Vector field UAV guidance for path following and obstacle avoidance with minimal deviation," *Journal of Guidance, Control, and Dynamics*, vol. 42, no. 8, pp. 1848–1856, 2019, doi: 10.2514/1. G004053.

[80] Y. Liu, Q. Wang, H. Hu, and Y. He, "A Novel Real-Time Moving Target Tracking and Path Planning System for a Quadrotor UAV in Unknown Unstructured Outdoor Scenes," *IEEE Trans Syst Man Cybern Syst*, vol. 49, no. 11, pp. 2362–2372, 2019, doi: 10.1109/TSMC.2018.2808471.

[81] B. Kovács, G. Szayer, F. Tajti, M. Burdelis, and P. Korondi, "A novel potential field method for path planning of mobile robots by adapting animal motion attributes," *Rob Auton Syst*, vol. 82, pp. 24–34, 2016, doi: 10.1016/j.robot.2016.04.007.

[82] E. Rimon and D. E. Koditschek, "Exact Robot Navigation using Artificial Potential Functions," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 5, pp. 501–518, 1992, doi: 10.1109/70.163777.

[83] L. De Filippis, G. Guglieri, and F. Quagliotti, “A minimum risk approach for path planning of UAVs,” *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 61, no. 1–4, pp. 203–219, 2011, doi: 10.1007/s10846-010-9493-9.

[84] S. Feyzabadi and S. Carpin, “Risk-aware path planning using hirerachical constrained Markov Decision Processes,” in *IEEE International Conference on Automation Science and Engineering*, IEEE, 2014, pp. 297–303. doi: 10.1109/CoASE.2014.6899341.

[85] S. Primatesta, G. Guglieri, and A. Rizzo, “A Risk-Aware Path Planning Strategy for UAVs in Urban Environments,” *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 95, no. 2, pp. 629–643, 2019, doi: 10.1007/s10846-018-0924-3.

[86] C. Ryan, F. Murphy, and M. Mullins, “Spatial risk modelling of behavioural hotspots: Risk-aware path planning for autonomous vehicles,” *Transp Res Part A Policy Pract*, vol. 134, no. January, pp. 152–163, 2020, doi: 10.1016/j.tra.2020.01.024.

[87] J. Da Silva Arantes, M. Da Silva Arantes, C. F. M. Toledo, O. T. Júnior, and B. C. Williams, “Heuristic and Genetic Algorithm Approaches for UAV Path Planning under Critical Situation,” *International Journal on Artificial Intelligence Tools*, vol. 26, no. 1, pp. 1–30, 2017, doi: 10.1142/S0218213017600089.

[88] E. Rudnick-Cohen, S. Azarm, and J. W. Herrmann, “Planning unmanned aerial system (UAS) takeoff trajectories to minimize third-party risk,” in *2019 International Conference on Unmanned Aircraft Systems, ICUAS 2019*, IEEE, 2019, pp. 1306–1315. doi: 10.1109/ICUAS.2019.8798149.

[89] X. Zhu, R. Yan, R. Peng, and Z. Zhang, “Optimal routing, loading and aborting of UAVs executing both visiting tasks and transportation tasks,” *Reliab Eng Syst Saf*, vol. 204, no. March, p. 107132, 2020, doi: 10.1016/j.ress.2020.107132.

[90] J. J. Chung, A. J. Smith, R. Skeele, and G. A. Hollinger, “Risk-aware graph search with dynamic edge cost discovery,” *International Journal of Robotics Research*, vol. 38, no. 2–3, pp. 182–195, 2019, doi: 10.1177/0278364918781009.

[91] E. Rudnick-Cohen, J. W. Herrmann, and S. Azarm, “Risk-based path planning optimization methods for unmanned aerial vehicles over inhabited areas,” *J Comput Inf Sci Eng*, vol. 16, no. 2, pp. 1–7, 2016, doi: 10.1115/1.4033235.

[92] N. Kim and Y. Yoon, “Cooperative sUAV Collision Avoidance Based on Satisficing Theory,” *International Journal of Aeronautical and Space Sciences*, vol. 20, no. 4, pp. 978–986, 2019, doi: 10.1007/s42405-019-00183-4.

[93] P. P. Y. Wu, D. Campbell, and T. Merz, "Multi-objective four-dimensional vehicle motion planning in large dynamic environments," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 41, no. 3, pp. 621–634, 2011, doi: 10.1109/TSMCB.2010.2061225.

[94] Y. Liu, X. Zhang, Y. Zhang, and X. Guan, "Collision free 4D path planning for multiple UAVs based on spatial refined voting mechanism and PSO approach," *Chinese Journal of Aeronautics*, vol. 32, no. 6, pp. 1504–1519, 2019, doi: 10.1016/j.cja.2019.03.026.

[95] V. Roberge, M. Tarbouchi, and G. Labonte, "Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning," *IEEE Trans Industr Inform*, vol. 9, no. 1, pp. 132–141, 2013, doi: 10.1109/TII.2012.2198665.

[96] S. Ragi and E. K. P. Chong, "UAV path planning in a dynamic environment via partially observable markov decision process," *IEEE Trans Aerosp Electron Syst*, vol. 49, no. 4, pp. 2397–2412, 2013, doi: 10.1109/TAES.2013.6621824.

[97] K. Dorling, J. Heinrichs, G. G. Messier, and S. Magierowski, "Vehicle Routing Problems for Drone Delivery," *IEEE Trans Syst Man Cybern Syst*, vol. 47, no. 1, pp. 70–85, 2017, doi: 10.1109/TSMC.2016.2582745.

[98] K. Peng, J. Du, F. Lu, Q. Sun, Y. Dong, P. Zhou and M Hu, "A Hybrid Genetic Algorithm on Routing and Scheduling for Vehicle-Assisted Multi-Drone Parcel Delivery," *IEEE Access*, vol. 7, pp. 49191–49200, 2019, doi: 10.1109/ACCESS.2019. 2910134.

[99] H. Qin, Z. Meng, W. Meng, X. Chen, H. Sun, F. Lin and M.H Ang, "Autonomous Exploration and Mapping System Using Heterogeneous UAVs and UGVs in GPS-Denied Environments," *IEEE Trans Veh Technol*, vol. 68, no. 2, pp. 1339–1350, 2019, doi: 10.1109/TVT.2018.2890416.

[100] D. Sacramento, D. Pisinger, and S. Ropke, "An adaptive large neighborhood search metaheuristic for the vehicle routing problem with drones," *Transp Res Part C Emerg Technol*, vol. 102, no. March 2018, pp. 289–315, 2019, doi: 10.1016/j.trc.2019.02.018.

[101] Q. Ning, G. Tao, B. Chen, Y. Lei, H. Yan, and C. Zhao, "Multi-UAVs trajectory and mission cooperative planning based on the Markov model," *Physical Communication*, vol. 35, p. 100717, 2019, doi: 10.1016/j.phycom.2019.100717.

[102] C. C. Murray and R. Raj, "The multiple flying sidekicks traveling salesman problem: Parcel delivery with multiple drones," *Transp Res Part C Emerg Technol*, vol. 110, no. October 2019, pp. 368–398, 2020, doi: 10.1016/j.trc.2019.11.003.

[103] M. Da Silva Arantes, C. F. M. Toledo, B. C. Williams, and M. Ono, "Collision-Free Encoding for Chance-Constrained Nonconvex Path Planning," *IEEE Transactions on Robotics*, vol. 35, no. 2, pp. 433–448, 2019, doi: 10.1109/TRO.2018.2878996.

[104] R. Prakash, R. Piplani, and J. Desai, "An optimal data-splitting algorithm for aircraft scheduling on a single runway to maximize throughput," *Transp Res Part C Emerg Technol*, vol. 95, no. August, pp. 570–581, 2018, doi: 10.1016/j.trc.2018.07.031.

[105] W. Ding, Y. Zhang, and M. Hansen, "Downstream impact of flight rerouting," *Transp Res Part C Emerg Technol*, vol. 88, no. June 2017, pp. 176–186, 2018, doi: 10.1016/j.trc.2018.01.010.

[106] Y. Zhang, R. Su, G. G. N. Sandamali, Y. Zhang, C. G. Cassandras, and L. Xie, "A Hierarchical Heuristic Approach for Solving Air Traffic Scheduling and Routing Problem with a Novel Air Traffic Model," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 9, pp. 3421–3434, 2019, doi: 10.1109/TITS.2018.2874235.

[107] J. Huang, H. Wang, Y. Qian, and C. Wang, "Priority-based traffic scheduling and utility optimization for cognitive radio communication infrastructure-based smart grid," *IEEE Trans Smart Grid*, vol. 4, no. 1, pp. 78–86, 2013, doi: 10.1109/TSG.2012.2227282.

[108] A. M. Ham, "Integrated scheduling of m-truck, m-drone, and m-depot constrained by time-window, drop-pickup, and m-visit using constraint programming," *Transp Res Part C Emerg Technol*, vol. 91, no. March, pp. 1–14, 2018, doi: 10.1016/j.trc.2018.03.025.

[109] W. Yao, N. Qi, N. Wan, and Y. Liu, "An iterative strategy for task assignment and path planning of distributed multiple unmanned aerial vehicles," *Aerosp Sci Technol*, vol. 86, pp. 455–464, 2019, doi: 10.1016/j.ast.2019.01.061.

[110] H. Tang, Y. Zhang, V. Mohmoodian, and H. Charkhgard, "Automated flight planning of high-density urban air mobility," *Transp Res Part C Emerg Technol*, vol. 131, no. August, p. 103324, 2021, doi: 10.1016/j.trc.2021.103324.

[111] F. Cheng, S. Zhang, Z. Li, Y. Chen, N. Zhao, F.R. Yu and V.C Leung, "UAV Trajectory Optimization for Data Offloading at the Edge of Multiple Cells," *IEEE Trans Veh Technol*, vol. 67, no. 7, pp. 6732–6736, 2018, doi: 10.1109/TVT.2018.2811942.

[112] M. Hu, W. Liu, K. Peng, X. Ma, W. Cheng, J. Liu and B. Li, "Joint routing and scheduling for vehicle-assisted multidrone surveillance," *IEEE Internet Things J*, vol. 6, no. 2, pp. 1781–1790, 2019, doi: 10.1109/JIOT.2018.2878602.

[113] L. O. Seman, L. A. Koehler, E. Camponogara, and W. Kraus, “Integrated headway and bus priority control in transit corridors with bidirectional lane segments,” *Transp Res Part C Emerg Technol*, vol. 111, no. December 2019, pp. 114–134, 2020, doi: 10.1016/j.trc.2019.12.001.

[114] X. Tang, P. Chen, and Y. Zhang, “4D trajectory estimation based on nominal flight profile extraction and airway meteorological forecast revision,” *Aerosp Sci Technol*, vol. 45, pp. 387–397, 2015, doi: 10.1016/j.ast.2015.06.001.

[115] S. Yan and K. Cai, “A multi-objective multi-memetic algorithm for network-wide conflict-free 4D flight trajectories planning,” *Chinese Journal of Aeronautics*, vol. 30, no. 3, pp. 1161–1173, 2017, doi: 10.1016/j.cja.2017.03.008.

[116] S. Hao, S. Cheng, and Y. Zhang, “A multi-aircraft conflict detection and resolution method for 4-dimensional trajectory-based operation,” *Chinese Journal of Aeronautics*, vol. 31, no. 7, pp. 1579–1593, 2018, doi: 10.1016/j.cja.2018.04.017.

[117] N. Takeichi, “Adaptive prediction of flight time uncertainty for ground-based 4D trajectory management,” *Transp Res Part C Emerg Technol*, vol. 95, pp. 335–345, 2018, doi: 10.1016/j.trc.2018.07.028.

[118] Q. Tan, Z. Wang, Y. S. Ong, and K. H. Low, “Evolutionary optimization-based mission planning for UAS traffic management (UTM),” *2019 International Conference on Unmanned Aircraft Systems, ICUAS 2019*, pp. 952–958, 2019, doi: 10.1109/ICUAS.2019.8798078.

[119] S. J. Kim and G. J. Lim, “A Real-Time Rerouting Method for Drone Flights Under Uncertain Flight Time,” *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 100, no. 3–4, pp. 1355–1368, 2020, doi: 10.1007/s10846-020-01214-z.

[120] C. Murray and M. Karwan, “A branch-and-bound-based solution approach for dynamic rerouting of airborne platforms,” *Naval Research Logistics*, vol. 60, no. 2, pp. 141–159, 2013, doi: 10.1002/nav.21526.

[121] J. Seo, Y. Kim, S. Kim, and A. Tsourdos, “Collision Avoidance Strategies for Unmanned Aerial Vehicles in Formation Flight,” *IEEE Trans Aerosp Electron Syst*, vol. 53, no. 6, pp. 2718–2734, 2017, doi: 10.1109/TAES.2017.2714898.

[122] A. Marchidan and E. Bakolas, “Collision avoidance for an unmanned aerial vehicle in the presence of static and moving obstacles,” *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 1, pp. 96–110, 2020, doi: 10.2514/1. G004446.

[123] I. K. Nikolos, K. P. Valavanis, N. C. Tsourveloudis, and A. N. Kostaras, "Evolutionary Algorithm Based Offline/Online Path Planner for UAV Navigation," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 33, no. 6, pp. 898–912, 2003, doi: 10.1109/TSMCB.2002.804370.

[124] X. Li, J. Zhang, J. Bian, Y. Tong, and T. Y. Liu, "A cooperative multi-agent reinforcement learning framework for resource balancing in complex logistics network," *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, vol. 2, pp. 980–988, 2019.

[125] C. Wang, J. Wang, Y. Shen, and X. Zhang, "Autonomous Navigation of UAVs in Large-Scale Complex Environments: A Deep Reinforcement Learning Approach," *IEEE Trans Veh Technol*, vol. 68, no. 3, pp. 2124–2136, 2019, doi: 10.1109/TVT.2018.2890773.

[126] M. Liu, H. Ma, J. Li, and S. Koenig, "Task and path planning for multi-agent pickup and delivery," *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, vol. 2, pp. 1152–1160, 2019.

[127] U. Challita, W. Saad, and C. Bettstetter, "Interference management for cellular-connected UAVs: A deep reinforcement learning approach," *IEEE Trans Wirel Commun*, vol. 18, no. 4, pp. 2125–2140, 2019, doi: 10.1109/TWC.2019.2900035.

[128] T. T. Nguyen, N. D. Nguyen, and S. Nahavandi, "Deep reinforcement learning for multi-agent systems: A review of challenges, solutions and applications," *ArXiv*, vol. 50, no. 9, pp. 3826–3839, 2020.

[129] T. I. Zohdi, "The Game of Drones: rapid agent-based machine-learning models for multi-UAV path planning," *Comput Mech*, vol. 65, no. 1, pp. 217–228, 2020, doi: 10.1007/s00466-019-01761-9.

[130] X. C. Ding, A. R. Rahmani, and M. Egerstedt, "Multi-UAV convoy protection: An optimal approach to path planning and coordination," *IEEE Transactions on Robotics*, vol. 26, no. 2, pp. 256–268, 2010, doi: 10.1109/TRO.2010.2042325.

[131] J. Hu, H. Zhang, L. Song, R. Schober, and H. V. Poor, "Cooperative Internet of UAVs: Distributed Trajectory Design by Multi-Agent Deep Reinforcement Learning," *IEEE Transactions on Communications*, vol. 68, no. 11, pp. 6807–6821, 2020, doi: 10.1109/TCOMM.2020.3013599.

[132] L. Wang, K. Wang, C. Pan, W. Xu, N. Aslam, and L. Hanzo, "Multi-Agent Deep Reinforcement learning based trajectory planning for multi-UAV assisted mobile edge computing," *IEEE Trans Cogn Commun Netw*, vol. 7, no. 1, pp. 73–84, 2021.

[133] Y. Mualla, A. Najjar, A. Daoud, S. Galland, C. Nicolle and E. Shakshuki, "Agent-based simulation of unmanned aerial vehicles in civilian applications: A systematic literature review and research directions," *Future Generation Computer Systems*, vol. 100, pp. 344–364, 2019, doi: 10.1016/j.future.2019.04.051.

[134] Roger and Schaufele, "FAA Aerospace Forecast Fiscal Years (2017-2037)," 2016.

[135] FAA and NASA, "FAA Unmanned Aircraft Systems Traffic Management (UTM) Concept of Operations - v1.0," 2018. [Online]. Available: https://utm.arc.nasa.gov/docs/2018-UTM-ConOps-v1.0.pdf

[136] FAA and NASA, "FAA UTM Concept of Operations - v2.0," 2020.

[137] S. Bradford, "Concept of Operations for Urban Air Mobility," *The FAA NextGen Office*, vol. 1, no. 1, pp. 1–49, 2020, [Online]. Available: https://nari.arc.nasa.gov/sites/default/files/attachments/UAM_ConOps_v1.0.pdf

[138] SESAR, "European Drones Outlook Study," 2016.

[139] SESAR, "U-space blueprint - SESAR Joint Undertaking," 2017. doi: 10.2829/614891.

[140] A. Donkels, "Trajectory Risk Evaluation for Autonomous Low-Flying Air Transport," *Journal of Guidance, Control, and Dynamics*, pp. 1–8, 2020, doi: 10.2514/1. g004678.

[141] B. Pang, K. H. Low, and C. Lv, "Adaptive conflict resolution for multi-UAV 4D routes optimization using stochastic fractal search algorithm," *Transp Res Part C Emerg Technol*, 2022, doi: 10.1016/j.trc.2022.103666.

[142] W. Dai, B. Pang, and K. H. Low, "Conflict-free four-dimensional path planning for urban air mobility considering airspace occupancy," *Aerosp Sci Technol*, vol. 119, p. 107154, 2021, doi: 10.1016/j.ast.2021.107154.

[143] MFB Mohamed Salleh, C Wanchao, Z Wang, S Huang, DY Tan, T Huang, KH Low, "Preliminary concept of adaptive urban airspace management for unmanned aircraft operations," *AIAA Information Systems-AIAA Infotech at Aerospace, 2018*, no. 209989, pp. 1–12, 2018, doi: 10.2514/6.2018-2260.

[144] C. H. J. Wang, E. M. Ng, and Kin Huat Low, "Investigation and Modeling of Flight Technical Error (FTE) Associated with UAS Operating with and without Pilot Guidance," *IEEE Trans Veh Technol*, pp. 1–13, 2021, doi: 10.1109/TVT.2021.3117081.

[145] Federal Aviation Administration, "Fact Sheet – Small Unmanned Aircraft Systems (UAS) Regulations (Part 107)," *U.S Department of Transportation*, 2020. https://www.faa.gov/news/fact_sheets/news_story.cfm?newsId=22615

[146] M. Izdebski, I. Jacyna-Gołda, and P. Gołda, “Minimisation of the probability of serious road accidents in the transport of dangerous goods,” *Reliab Eng Syst Saf*, vol. 217, no. September 2021, p. 108093, 2021, doi: 10.1016/j.ress.2021.108093.

[147] Y. Zou, H. Zhang, G. Zhong, H. Liu, and D. Feng, “Collision probability estimation for small, unmanned aircraft systems,” *Reliab Eng Syst Saf*, vol. 213, no. July 2020, p. 107619, 2021, doi: 10.1016/j.ress.2021.107619.

[148] R.M.A. Valdés, V.F.G. Comendador, J.A.P. Castán, A. R. Sanz, L. P. Sanz, E. S. Ayra, and F. J. S. Nieto, “Development of safety performance functions (SPFs) to analyse and predict aircraft loss of separation in accordance with the characteristics of the airspace,” *Reliab Eng Syst Saf*, vol. 186, no. February, pp. 143–161, 2019, doi: 10.1016/j.ress.2019.02.007.

[149] Q. M. Ha, Y. Deville, Q. D. Pham, and M. H. Hà, “On the min-cost Traveling Salesman Problem with Drone,” *Transp Res Part C Emerg Technol*, vol. 86, no. November 2017, pp. 597–621, 2018, doi: 10.1016/j.trc.2017.11.015.

[150] J. Cho and Y. Yoon, “Extraction and interpretation of geometrical and topological properties of urban airspace for UAS operations,” in *13th USA/Europe Air Traffic Management Research and Development Seminar 2019*, 2019.

[151] S. Bertrand, N. Raballand, F. Viguier, and F. Muller, “Ground risk assessment for long-range inspection missions of railways by UAVs,” in *2017 International Conference on Unmanned Aircraft Systems, ICUAS 2017*, 2017, pp. 1343–1351. doi: 10.1109/ICUAS. 2017.7991331.

[152] S. Primatesta, A. Rizzo, and A. la Cour-Harbo, “Ground Risk Map for Unmanned Aircraft in Urban Environments,” *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 97, no. 3–4, pp. 489–509, 2020, doi: 10.1007/s10846-019-01015-z.

[153] J. Rappaport, “Consumption amenities and city population density,” *Reg Sci Urban Econ*, vol. 38, no. 6, pp. 533–552, 2008, doi: 10.1016/j.regsciurbeco.2008.02.001.

[154] Y. Yao, X. Liu, X. Li, J. Zhang, Z. Liang, K. Mai, and Y. Zhang, “Mapping fine-scale population distributions at the building level by integrating multisource geospatial big data,” *International Journal of Geographical Information Science*, vol. 31, no. 6, pp. 1220–1244, 2017, doi: 10.1080/13658816.2017.1290252.

[155] P. Deville, C. Linard, S. Martin, M. Gilbert, F.R. Stevens, A.E. Gaughan, V.D Blondel. and A.J. Tatem, “Dynamic population mapping using mobile phone data,” in

Proceedings of the National Academy of Sciences of the United States of America, 2014, pp. 15888–15893. doi: 10.1073/pnas.1408439111.

[156] L. Ang and M. Hansen, “Clustering based approach to single drone path planning in complex urban airspace,” 2019.

[157] W. N. Alexander and J. Whelchel, “Flyover noise of multi-rotor sUAS,” in *INTER-NOISE 2019 MADRID - 48th International Congress and Exhibition on Noise Control Engineering*, 2019.

[158] V. Bulusu, V. Polishchuk, R. Sengupta, and L. Sedov, “Capacity Estimation for Low Altitude Airspace,” in *17th AIAA Aviation Technology, Integration, and Operations Conference*, 2017. doi: 10.2514/6.2017-4266.

[159] J. Lee and M. Mitici, “An integrated assessment of safety and efficiency of aircraft maintenance strategies using agent-based modelling and stochastic Petri nets,” *Reliab Eng Syst Saf*, vol. 202, no. May, p. 107052, 2020, doi: 10.1016/j.ress.2020.107052.

[160] P. E. Hart, N. J. Nilsson, and B. Raphael, “A Formal Basis for the Heuristic Determination of Minimum Cost Paths,” *IEEE Transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968, doi: 10.1109/TSSC.1968.300136.

[161] T. C. Morphology, “Expanding from Discrete to Continuous Estimation of Distribution Algorithms: The IDEA,” in *International Conference on Parallel Problem Solving from Nature*, 2000.

[162] Q. Zhang, J. Sun, E. Tsang, and J. Ford, “Hybrid estimation of distribution algorithm for global optimization,” in *Engineering Computations (Swansea, Wales)*, 2004, pp. 91–107. doi: 10.1108/02644400410511864.

[163] G. Pauer and Á. Török, “Introducing a novel safety assessment method through the example of a reduced complexity binary integer autonomous transport model,” *Reliab Eng Syst Saf*, vol. 217, no. September 2021, 2022, doi: 10.1016/j.ress.2021.108062.

[164] WorldoMeter, “The population density in Singapore,” *www.Worldometers.info*, 2021.

[165] SG Land Transport Authority, “Singapore Road Traffic Conditions during Peak Hours,” *Singapore Open Data Licence*, 2021.

[166] Budget Direct Insurance, “Road Accident Statistics Singapore 2021,” *Land Transport Authority*, 2021.

[167] Y. Wu, K. H. Low, B. Pang, and Q. Tan, "Swarm-based 4D Path Planning for Drone Operations in Urban Environments," *IEEE Trans Veh Technol*, vol. 9545, no. c, 2021, doi: 10.1109/TVT.2021.3093318.

[168] Wikipedia, "List of cities proper by population density," 2021. https://en.wikipedia.org/wiki/List_of_cities_proper_by_population_density

[169] A. Li and M. Hansen, "Obstacle Clustering and Path Optimization for Drone Routing," in *9th International Conference for Research in Air Transportation*, 2020, pp. 1–8.

[170] B. Pang, X. Hu, W. Dai, and K. H. Low, "UAV path optimization with an integrated cost assessment model considering third-party risks in metropolitan environments," *Reliab Eng Syst Saf*, vol. 222, no. February, p. 108399, 2022, doi: 10.1016/j.ress.2022.108399.

[171] K. Wang and A. Jacquillat, "A stochastic integer programming approach to air traffic scheduling and operations," *Oper Res*, vol. 68, no. 5, pp. 1375–1402, Sep. 2020, doi: 10.1287/opre.2020.1985.

[172] D. Yu, A. G. Ebadi, K. Jermsittiparsert, N. H. Jabarullah, M. V. Vasiljeva, and S. Nojavan, "Risk-Constrained Stochastic Optimization of a Concentrating Solar Power Plant," *IEEE Trans Sustain Energy*, vol. 11, no. 3, pp. 1464–1472, Jul. 2020, doi: 10.1109/TSTE.2019.2927735.

[173] X. Z. Xu, Y. F. Niu, and Y. F. Song, "Computing the reliability of a stochastic distribution network subject to budget constraint," *Reliab Eng Syst Saf*, vol. 216, no. August, p. 107947, 2021, doi: 10.1016/j.ress.2021.107947.

[174] W. B. Powell, "A unified framework for stochastic optimization," *European Journal of Operational Research*, vol. 275, no. 3. Elsevier B.V., pp. 795–821, Jun. 16, 2019. doi: 10.1016/j.ejor.2018.07.014.

[175] D. Bertsimas and N. Youssef, "Stochastic optimization in supply chain networks: averaging robust solutions," *Optim Lett*, vol. 14, no. 4, pp. 839–855, Jun. 2020, doi: 10.1007/s11590-019-01405-0.

[176] Y. Huang, L. Zhao, T. van Woensel, and J. P. Gross, "Time-dependent vehicle routing problem with path flexibility," *Transportation Research Part B: Methodological*, vol. 95, pp. 169–195, Jan. 2017, doi: 10.1016/j.trb.2016.10.013.

[177] Y. Hong, B. Choi, and Y. Kim, "Two-Stage Stochastic Programming Based on Particle Swarm Optimization for Aircraft Sequencing and Scheduling," *IEEE Transactions on*

Intelligent Transportation Systems, vol. 20, no. 4, pp. 1365–1377, Apr. 2019, doi: 10.1109/TITS.2018.2850000.

[178] A. Khassiba, S. Cafieri, F. Bastin, M. Mongeau, and B. Gendron, "Two-stage stochastic programming models for the extended aircraft arrival management problem with multiple pre-scheduling points," *Transp Res Part C Emerg Technol*, vol. 142, Sep. 2022, doi: 10.1016/j.trc.2022.103769.

[179] N. N. Patel, Ã. F. R. Stevens, Z. Huang, A. E. Gaughan, I. Elyazar, and A. J. Tatem, "Improving Large Area Population Mapping Using Geotweet Densities," *Transactions in GIS*, vol. 21, no. 2, pp. 317–331, 2017, doi: 10.1111/tgis.12214.

[180] Z. Cheng, M. Trépanier, and L. Sun, "Real-Time Forecasting of Metro Origin-Destination Matrices with High-Order Weighted Dynamic Mode Decomposition," *Transportation Science*, vol. 56, no. 4, pp. 904–918, Jul. 2022, doi: 10.1287/trsc.2022.1128.

[181] Z. Cheng, M. Trépanier, and L. Sun, "Incorporating travel behavior regularity into passenger flow forecasting," *Transp Res Part C Emerg Technol*, vol. 128, Jul. 2021, doi: 10.1016/j.trc.2021.103200.

[182] B. Mo, Y. Shen, and J. Zhao, "Impact of Built Environment on First- and Last-Mile Travel Mode Choice," *Transp Res Rec*, vol. 2672, no. 6, pp. 40–51, Dec. 2018, doi: 10.1177/0361198118788423.

[183] M. Rosenblatt, "The Harmonic Analysis of Bounded Functions," *Duke Math. J*, vol. 20, p. 146, 1953, [Online]. Available: https://www.pnas.org

[184] Z. Wang and J. B. Sheu, "Vehicle routing problem with drones," *Transportation Research Part B: Methodological*, vol. 122, pp. 350–364, 2019, doi: 10.1016/j.trb.2019.03.005.

[185] C. Cheng, Y. Adulyasak, and L. M. Rousseau, "Drone routing with energy function: Formulation and exact algorithm," *Transportation Research Part B: Methodological*, vol. 139, pp. 364–387, 2020, doi: 10.1016/j.trb.2020.06.011.

[186] Y. Wu, K. H. Low, and X. Hu, "Trajectory-based flight scheduling for AirMetro in urban environments by conflict resolution," *Transp Res Part C Emerg Technol*, vol. 131C, pp. 1–56, 2021.

[187] R. She and Y. Ouyang, "Efficiency of UAV-based last-mile delivery under congestion in low-altitude air," *Transp Res Part C Emerg Technol*, vol. 122, no. June 2020, p. 102878, 2021, doi: 10.1016/j.trc.2020.102878.

[188] D. Bertsimas and S. S. Patterson, "The air traffic flow management problem with enroute capacities," *Oper Res*, vol. 46, no. 3, pp. 406–422, 1998, doi: 10.1287/opre.46.3.406.

[189] Benders J, "Partitioning procedures for solving mixed-variable programming problems," *Numer Math (Heidelb)*, vol. 4, no. 1, pp. 238–252, 1962.

[190] G. Laporte and F. v Louveaux, "The integer L-shaped method for stochastic integer programs with complete recourse," 1993.

[191] S. Ahmed, M. Tawarmalani, and N. v. Sahinidis, "A finite branch-and-bound algorithm for two-stage stochastic integer programs," *Math Program*, vol. 100, no. 2, pp. 355–377, Jun. 2004, doi: 10.1007/s10107-003-0475-6.

[192] S. Sen and J. L. Higle, "The C3 theorem and a D2 algorithm for large scale stochastic mixed-integer programming: Set convexification," *Mathematical Programming*, vol. 104, no. 1. pp. 1–20, Sep. 2005. doi: 10.1007/s10107-004-0566-z.

[193] J. Zou, S. Ahmed, and X. A. Sun, "Stochastic dual dynamic integer programming," *Math Program*, vol. 175, no. 1–2, pp. 461–502, May 2019, doi: 10.1007/s10107-018-1249-5.

[194] H. Salimi, "Stochastic Fractal Search: A powerful metaheuristic algorithm," *Knowl Based Syst*, vol. 75, pp. 1–18, 2015, doi: 10.1016/j.knosys.2014.07.025.

[195] R. Cheng and Y. Jin, "A social learning particle swarm optimization algorithm for scalable optimization," *Inf Sci (N Y)*, vol. 291, no. C, pp. 43–60, 2015, doi: 10.1016/j.ins.2014.08.039.

[196] FAA, "UAS Center of Excellence Task A4: UAS Ground Collision Severity Evaluation," 2017. Accessed: Dec. 21, 2022. [Online]. Available: https://www.skybrary.aero/sites/default/files/bookshelf/4073.pdf

[197] Y. Zhao, Z. Zheng, and Y. Liu, "Survey on computational-intelligence-based UAV path planning," *Knowl Based Syst*, vol. 158, no. March, pp. 54–64, 2018, doi: 10.1016/j.knosys.2018.05.033.

[198] I. C. Kleinbekman, M. Mitici, and P. Wei, "Rolling-horizon electric vertical takeoff and landing arrival scheduling for on-demand urban air mobility," *Journal of Aerospace Information Systems*, vol. 17, no. 3, pp. 150–159, 2020, doi: 10.2514/1.I010776.

[199] E. S. Rigas, P. Kolios, and G. Ellinas, “Scheduling Aerial Vehicles in an Urban Air Mobility Scheme,” 2021.

[200] O. Rodionova, D. Delahaye, B. Sridhar, and H. K. Ng, “Deconflicting Wind-Optimal Aircraft Trajectories in North Atlantic Oceanic Airspace,” in *Advanced Aircraft Efficiency in a Global Air Transport System Conference (AEGATS)*, 2016, pp. 1–12.

[201] S. Birolini, A. P. Antunes, M. Cattaneo, P. Malighetti, and S. Paleari, “Integrated flight scheduling and fleet assignment with improved supply-demand interactions,” *Transportation Research Part B: Methodological*, vol. 149, pp. 162–180, 2021, doi: 10.1016/j.trb.2021.05.001.

[202] T. M. Cabreira, C. Di Franco, P. R. Ferreira, and G. C. Buttazzo, “Energy-Aware spiral coverage path planning for UAV photogrammetric applications,” *IEEE Robot Autom Lett*, vol. 3, no. 4, pp. 3662–3668, 2018, doi: 10.1109/LRA.2018.2854967.

[203] N. Peng, Y. Xi, J. Rao, X. Ma, and F. Ren, “Urban Multiple Route Planning Model Using Dynamic Programming in Reinforcement Learning,” *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, 2021, doi: 10.1109/TITS.2021.3075221.

[204] K. D. Bilimoria, “A geometric optimization approach to aircraft conflict resolution,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Denver, CO, 2000. doi: 10.2514/6.2000-4265.

[205] D. Rey, C. Rapine, R. Fondacci, and N. E. El Faouzi, “Subliminal speed control in air traffic management: Optimization and simulation,” *Transportation Science*, vol. 50, no. 1, pp. 240–262, 2016, doi: 10.1287/trsc.2015.0602.

[206] V. D. Berdonosov, A. A. Zivotova, D. O. Zhuravlev, and Z. H. Naing, “Implementation of the Speed Approach for UAV Collision Avoidance in Dynamic Environment,” *2018 International Multi-Conference on Industrial Engineering and Modern Technologies, FarEastCon 2018*, 2018, doi: 10.1109/FarEastCon.2018.8602815.

[207] X. Geng, Y. Wang, P. Wang, and B. Zhang, “Motion plan of maritime autonomous surface ships by dynamic programming for collision avoidance and speed optimization,” *Sensors (Switzerland)*, vol. 19, no. 2, 2019, doi: 10.3390/s19020434.

[208] W. Dai, B. Pang, and K. H. Low, “Conflict-Free Four-Dimensional Path Planning for Urban Air Mobility Considering Airspace Occupations,” *Aerosp Sci Technol*, vol. 119, p. 107154, 2021, doi: 10.1016/j.ast.2021.107154.

[209] M. Pelegrín and C. D'Ambrosio, "Aircraft Deconfliction via Mathematical Programming: Review and Insights," *Transportation Science*, vol. 56, no. 1, pp. 118–140, 2022, doi: 10.1287/trsc.2021.1056.

[210] E. Hernández-Romero, A. Valenzuela, and D. Rivas, "Probabilistic multi-aircraft conflict detection and resolution considering wind forecast uncertainty," *Aerosp Sci Technol*, vol. 105, p. 105973, 2020, doi: 10.1016/j.ast.2020.105973.

[211] V. Courchelle, M. Soler, D. González-Arribas, and D. Delahaye, "A simulated annealing approach to 3D strategic aircraft deconfliction based on en-route speed changes under wind and temperature uncertainties," *Transp Res Part C Emerg Technol*, vol. 103, no. March, pp. 194–210, 2019, doi: 10.1016/j.trc.2019.03.024.

[212] Y. Wu, K. H. Low, and C. Lv, "Cooperative Path Planning for Heterogeneous Unmanned Vehicles in a Search-and-Track Mission Aiming at an Underwater Target," *IEEE Trans Veh Technol*, vol. 69, no. 6, pp. 6782–6787, 2020, doi: 10.1109/TVT.2020.2991983.

[213] W. Ding, W. Gao, K. Wang, and S. Shen, "An Efficient B-Spline-Based Kinodynamic Replanning Framework for Quadrotors," *IEEE Transactions on Robotics*, vol. 35, no. 6, pp. 1287–1306, 2019, doi: 10.1109/TRO.2019.2926390.

[214] G. Kahn, A. Villaflor, V. Pong, P. Abbeel, and S. Levine, "Uncertainty-Aware Reinforcement Learning for Collision Avoidance," *ArXiv*, pp. 1–12, 2017.

[215] X. Yang and P. Wei, "Scalable multi-agent computational guidance with separation assurance for autonomous urban air mobility," *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 8, pp. 1473–1486, 2020, doi: 10.2514/1.G005000.

[216] X. Yang and P. Wei, "Autonomous Free Flight Operations in Urban Air Mobility with Computational Guidance and Collision Avoidance," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–14, 2021, doi: 10.1109/TITS.2020.3048360.

[217] L. Pallottino, E. M. Feron, and A. Bicchi, "Conflict Resolution Problems for Air Traffic Management Systems Solved with Mixed Integer Programming," *IEEE Transactions on Intelligent Transportation Systems*, vol. 3, no. 1, pp. 3–11, 2002, doi: 10.1109/6979.994791.

[218] S. Burer and A. N. Letchford, "Non-convex mixed-integer nonlinear programming: A survey," *Surveys in Operations Research and Management Science*, vol. 17, no. 2, pp. 97–106, 2012, doi: 10.1016/j.sorms.2012.08.001.

[219] S. Arora, “Approximation schemes for geometric NP-hard problems: A survey,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 2245, pp. 16–17, 2001, doi: 10.1007/3-540-45294-x_2.

[220] N. Durand and J. M. Alliot, “Ant Colony Optimization for air traffic conflict resolution,” *Proceedings of the 8th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2009*, pp. 182–187, 2009.

[221] H. A. Amr, M. Khajehzadeh, M. Taha, Z. Beheshti, S. Mariyam, and H. Shamsuddin, “A Review of Population-based Meta- Heuristic Algorithm,” *Int. J. Advance. Soft Comput. Appl*, vol. 5, no. 1, pp. 2074–8523, 2013.

[222] P. Loubière, A. Jourdan, P. Siarry, and R. Chelouah, “A sensitivity analysis method aimed at enhancing the metaheuristics for continuous optimization,” *Artif Intell Rev*, vol. 50, no. 4, pp. 625–647, 2018, doi: 10.1007/s10462-017-9553-7.

[223] M. A. Mellal and E. Zio, “A penalty guided stochastic fractal search approach for system reliability optimization,” *Reliab Eng Syst Saf*, vol. 152, pp. 213–227, 2016, doi: 10.1016/j.ress.2016.03.019.

[224] S. Aras, E. Gedikli, and H. T. Kahraman, “A novel stochastic fractal search algorithm with fitness-Distance balance for global numerical optimization,” *Swarm Evol Comput*, vol. 61, no. December 2020, p. 100821, 2021, doi: 10.1016/j.swevo.2020.100821.

[225] K. C. Tan, S. C. Chiam, A. A. Mamun, and C. K. Goh, “Balancing exploration and exploitation with adaptive variation for evolutionary multi-objective optimization,” *Eur J Oper Res*, vol. 197, no. 2, pp. 701–713, 2009, doi: 10.1016/j.ejor.2008.07.025.

[226] E. Cuevas, A. Echavarría, and M. A. Ramírez-Ortegón, “An optimization algorithm inspired by the States of Matter that improves the balance between exploration and exploitation,” *Applied Intelligence*, vol. 40, no. 2, pp. 256–272, 2014, doi: 10.1007/s10489-013-0458-0.

[227] H. T. Kahraman, S. Aras, and E. Gedikli, “Fitness-distance balance (FDB): A new selection method for meta-heuristic search algorithms,” *Knowl Based Syst*, vol. 190, p. 105169, 2020, doi: 10.1016/j.knosys.2019.105169.

[228] DJI, “User Manual Phantom 4 PRO V1.6,” 2018.

[229] X. Yang and P. Wei, “Autonomous On-Demand Free Flight Operations in Urban Air Mobility using Monte Carlo Tree Search,” in *International Conference on Research in*

Air Transportation (ICRAT), Barcelona, Spain, 2018. [Online]. Available: https://www.aere.iastate.edu/~pwei/proceedings/icrat18_MCTS.pdf

[230] A. Evans, M. Egorov, and S. Campbell, "Accommodating Operational Uncertainty in Urban Air Mobility Operations with Strategic Deconfliction," in *AIAA AVIATION Forum 2021*, American Institute of Aeronautics and Astronautics Inc, AIAA, 2021. doi: 10.2514/6.2021-2333.

[231] FAA, "Concept of operation on Trajectory Based Operations (TBO), Feb 2022.," 2022. [Online]. Available: https://www.faa.gov/air_traffic/technology/tbo#:~:text=Trajectory

[232] D. V. Sasso, F. Djeumou Fomeni, G. Lulli, and K. G. Zografos, "Planning efficient 4D trajectories in Air Traffic Flow Management.," *Eur J Oper Res*, vol. 276, no. 2, pp. 676–687, 2019, doi: 10.1016/j.ejor.2019.01.039.

[233] N. Schefers, Juan José Ramos González, Pau Folch, and José Luis Munoz-Gamarra., "A constraint programming model with time uncertainty for cooperative flight departures," *Transp Res Part C Emerg Technol*, vol. 96, pp. 170–191, 2018, doi: 10.1016/j.trc.2018.09.013.

[234] C. Cheng, Y. Adulyasak, L.-M. Rousseau, and M. Sim, "Robust Drone Delivery with Weather Information," *History*, 2020.

[235] A. Montlaur and L. Delgado, "Delay assignment optimization strategies at pre-tactical and tactical levels," in *SESAR Innovation Days*, 2015.

[236] D. Bertsimas, G. Lulli, and A. Odoni, "An integer optimization approach to large-scale air traffic flow management," *Oper Res*, vol. 59, no. 1, pp. 211–227, Jan. 2011, doi: 10.1287/opre.1100.0899.

[237] J. Chen, L. Chen, and D. Sun, "Air traffic flow management under uncertainty using chance-constrained optimization," *Transportation Research Part B: Methodological*, vol. 102, pp. 124–141, Aug. 2017, doi: 10.1016/j.trb.2017.05.014.

[238] G. G. N. Sandamali, R. Su, and Y. Zhang, "Flight Routing and Scheduling under Departure and en Route Speed Uncertainty," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 5, pp. 1915–1928, May 2020, doi: 10.1109/TITS.2019.2907058.

[239] A. Dudoit, V. Rimša, and M. Bogdevičius, "Stochastic Dynamic Aircraft System Conflict Distribution under Uncertainties," *Entropy*, vol. 24, no. 5, May 2022, doi: 10.3390/e24050583.

[240] D. Şimşek and M. S. Aktürk, “Resilient airline scheduling to minimize delay risks,” *Transp Res Part C Emerg Technol*, vol. 141, Aug. 2022, doi: 10.1016/j.trc.2022.103734.

[241] A. Muñoz Hernández, M. Polaina, A. Güemes, M. F. Soler Arnedo, D. González Arribas, J. Pons Prats, X. Prats Menéndez, A. Tutku Altun, E. Koyuncu, A. Kuenz, R. Zopp, D. Delahaye, “Data-driven methodology for uncertainty quantification of aircraft trajectory predictions,” in *2021 IEEE/AIAA 40th Digital Avionics Systems Conference.*, San Antonio, TX, USA., 2021, pp. 1–10.

[242] Y. Pang, X. Zhao, H. Yan, and Y. Liu, “Data-driven trajectory prediction with weather uncertainties: A Bayesian deep learning approach,” *Transp Res Part C Emerg Technol*, vol. 130, Sep. 2021, doi: 10.1016/j.trc.2021.103326.

[243] J. Lee, L. Marla, and A. Jacquillat, “Dynamic Disruption Management in Airline Networks Under Airport Operating Uncertainty,” *Transportation Science*, vol. 54, no. 4, pp. 973–997, Jul. 2020, doi: 10.1287/trsc.2020.0983.

[244] F. Sun, H. Liu, and Y. Zhang, “Integrated Aircraft and Passenger Recovery with Enhancements in Modeling, Solution Algorithm, and Intermodalism,” *IEEE Transactions on Intelligent Transportation Systems*, Jul. 2021, doi: 10.1109/TITS.2021.3090329.

[245] J. M. Hoekstra and J. Ellerbroek, “Aerial Robotics: State-based Conflict Detection and Resolution (Detect and Avoid) in High Traffic Densities and Complexities,” *Current Robotics Reports*, vol. 2, p. 297307, 2021, doi: 10.1007/s43154-021-00061-6/Published.

[246] Z. Liu, J. Zhang, Y. Zhu, and K. Cai, “Hierarchical Four-Dimensional Trajectories Planning Method for Manned and Unmanned Aircraft Integrated Airspace,” *Journal of Guidance, Control, and Dynamics*, vol. 45, no. 6, pp. 1017–1032, 2022, doi: 10.2514/1.G006206.

[247] D. Y. Tan, S. Badrinath, and H. Balakrishnan, “Analysis and design of Trajectory-Based Operations under wind forecast uncertainty,” *CEAS Aeronaut J*, Apr. 2021, doi: 10.1007/s13272-021-00501-1.

[248] S. A. M. Shihab, P. Wei, D. S. J. Ramirez, R. Mesa-Arango, and C. Bloebaum, “By Schedule or On Demand? - A Hybrid Operation Concept for Urban Air Mobility,” no. June, pp. 1–13, 2019, doi: 10.2514/6.2019-3522.

[249] P. Wu, X. Yang, P. Wei, and J. Chen, “Safety Assured Online Guidance with Airborne Separation for Urban Air Mobility Operations in Uncertain Environments,” *IEEE*

Transactions on Intelligent Transportation Systems, 2022, doi: 10.1109/TITS.2022. 3163657.

[250] P. Wu, J. Xie, Y. Liu, and J. Chen, “Risk-bounded and fairness-aware path planning for urban air mobility operations under uncertainty,” *Aerosp Sci Technol*, vol. 127, p. 107738, Aug. 2022, doi: 10.1016/j.ast.2022.107738.

[251] R. Chai, D. Liu, T. Liu, A. Tsourdos, Y. Xia, and S. Chai, “Deep Learning-Based Trajectory Planning and Control for Autonomous Ground Vehicle Parking Maneuver,” *IEEE Transactions on Automation Science and Engineering*, 2022, doi: 10.1109/TASE. 2022.3183610.

[252] R. Chai, A. Tsourdos, A. Savvaris, S. Chai, Y. Xia, and C. L. Philip Chen, “Design and Implementation of Deep Neural Network-Based Control for Automatic Parking Maneuver Process,” *IEEE Trans Neural Netw Learn Syst*, vol. 33, no. 4, pp. 1400–1413, Apr. 2022, doi: 10.1109/TNNLS.2020.3042120.

[253] R. Chai, A. Tsourdos, S. Chai, Y. Xia, A. Savvaris, and C. L. P. Chen, “Multi-Phase Overtaking Maneuver Planning for Autonomous Ground Vehicles via a Desensitized Trajectory Optimization Approach,” *IEEE Trans Industr Inform*, Jan. 2022, doi: 10.1109/TII.2022.3168434.

[254] R. Chai, A. Tsourdos, A. Savvaris, S. Wang, Y. Xia, and S. Chai, “Fast Generation of Chance-Constrained Flight Trajectory for Unmanned Vehicles,” *IEEE Trans Aerosp Electron Syst*, vol. 57, no. 2, pp. 1028–1045, Apr. 2021, doi: 10.1109/TAES.2020. 3037417.

[255] X. Sun, B. Zhang, R. Chai, A. Tsourdos, and S. Chai, “UAV trajectory optimization using chance-constrained second-order cone programming,” *Aerosp Sci Technol*, vol. 121, Feb. 2022, doi: 10.1016/j.ast.2021.107283.

[256] R. Chai, A. Tsourdos, A. Savvaris, S. Chai, Y. Xia, and C. L. P. Chen, “Multiobjective Optimal Parking Maneuver Planning of Autonomous Wheeled Vehicles,” *IEEE Transactions on Industrial Electronics*, vol. 67, no. 12, pp. 10809–10821, Dec. 2020, doi: 10.1109/TIE.2019.2962482.

[257] R. Chai, A. Tsourdos, A. Savvaris, S. Chai, and Y. Xia, “Solving Constrained Trajectory Planning Problems Using Biased Particle Swarm Optimization,” *IEEE Trans Aerosp Electron Syst*, vol. 57, no. 3, pp. 1685–1701, Jun. 2021, doi: 10.1109/TAES.2021. 3050645.

[258] Z. Zhou, J. Chen, and Y. Liu, "Optimized Landing of Drones in the Context of Congested Air Traffic and Limited Vertiports," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 9, pp. 6007–6017, Sep. 2021, doi: 10.1109/TITS.2020.3040549.

[259] B. Du, J. Chen, D. Sun, S. G. Manyam, and D. W. Casbeer, "UAV Trajectory Planning with Probabilistic Geo-Fence via Iterative Chance-Constrained Optimization," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 5859–5870, Jun. 2022, doi: 10.1109/TITS.2021.3060377.

[260] C. Chin, K. Gopalakrishnan, M. Egorov, A. Evans, and H. Balakrishnan, "Efficiency and Fairness in Unmanned Air Traffic Flow Management," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 9, pp. 5939–5951, Sep. 2021, doi: 10.1109/TITS.2020.3048356.

[261] C. Deng, C. H. J. Wang, and K. H. Low, "Investigation of Using Sky Openness Ratio as Predictor for Navigation Performance in Urban-like Environment to Support PBN in UTM," *Sensors*, vol. 22, no. 3, Feb. 2022, doi: 10.3390/s22030840.

[262] J. Holden and N. Goel, "Fast-Forwarding to a Future of On-Demand Urban Air Transportation," 2016. [Online]. Available: https://www.uber.com/elevate.pdf

[263] W. Han, A. Jasour, and B. Williams, "Non-Gaussian Risk Bounded Trajectory Optimization for Stochastic Nonlinear Systems in Uncertain Environments," in *arXiv*, Mar. 2022. [Online]. Available: http://arxiv.org/abs/2203.03038

[264] D. Bertsimas and A. Thiele, "Robust and Data-Driven Optimization: Modern Decision Making Under Uncertainty," in *Models, Methods, and Applications for Innovative Decision Making*, INFORMS, 2006, pp. 95–122. doi: 10.1287/educ.1063.0022.

[265] N. Kafle and B. Zou, "Modeling flight delay propagation: A new analytical-econometric approach," *Transportation Research Part B: Methodological*, vol. 93, pp. 520–542, 2016, doi: 10.1016/j.trb.2016.08.012.

[266] V. D. Sasso, F. D. Fomeni, G. Lulli, and K. G. Zografos, "Incorporating Stakeholders' priorities and preferences in 4D trajectory optimization," *Transportation Research Part B: Methodological*, vol. 117, pp. 594–609, 2018, doi: 10.1016/j.trb.2018.09.009.

[267] P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, and A. Mahajan, *Mixed-integer nonlinear optimization*, vol. 22. 2013. doi: 10.1017/S0962492913000032.

[268] L. S. Jennings and K. L. Teo, "A computational algorithm for functional inequality constrained optimization problems," *Automatica*, vol. 26, no. 2, pp. 371–375, 1990.

[269] DJI, "Inspire 1 - Product Information," 2014. Accessed: Sep. 09, 2022. [Online]. Available: https://www.dji.com/sg/inspire-1/info#specs

[270] A. Straubinger, R. Rothfeld, M. Shamiyeh, K. D. Büchter, J. Kaiser, and K. O. Plötner, "An overview of current research and developments in urban air mobility – Setting the scene for UAM introduction," *J Air Transp Manag*, vol. 87, no. June 2020, doi: 10.1016/j.jairtraman.2020.101852.

[271] M. Patterson, "Advanced Air Mobility (AAM): An Overview and Brief History," in *Transportation Engineering and Safety Conference.*, 2021.

[272] P. Hang, C. Lv, C. Huang, Y. Xing, and Z. Hu, "Cooperative Decision Making of Connected Automated Vehicles at Multi-Lane Merging Zone: A Coalitional Game Approach," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–13, 2021, doi: 10.1109/TITS.2021.3069463.

[273] S. Li, M. Egorov, and M. J. Kochenderfer, "Optimizing Collision Avoidance in Dense Airspace using Deep Reinforcement Learning," in *Thirteenth USA/Europe Air Traffic Management Research and Development Seminar (ATM2019)*, 2019, pp. 1–10.