Cranfield UNIVERSITY

Sangjun Bae

Multiple Agents Routing and Scheduling Algorithms for Network-Based Transportation Systems

School of Aerospace, Transport and Manufacturing,

Centre for Autonomous and Cyber-physical Systems

PhD

Cranfield UNIVERSITY

School of Aerospace, Transport and Manufacturing

Multiple Agents Routing and Scheduling Algorithms for Network-Based Transportation Systems

Sangjun Bae

PhD Thesis

Academic Year 2014-2018

Supervisors: Dr Hyo-Sang Shin and Professor Antonios Tsourdos

October 2018

This thesis is submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

Abstract

This research attempts to develop effective and practical algorithms that enable multiple agents to address routing and scheduling problems simultaneously: given a set of initial points and final points for multiple agents in a route network, separation-compliant routes and speed profiles are to be found for every agent while maximising a performance index subject to satisfy operational constraints. The algorithms are applicable to many transportation systems that consider many operational factors such as flight planning problems in the Air Traffic Management (ATM) system, and analysing urban airspace structure for an Unmanned Aircraft System (UAS) Traffic Management (UTM) system.

This thesis focuses on an investigation of a new horizontal Routing and Scheduling (R&S) algorithm for homogeneous multiple arrivals at a single airport. Importantly, this study is the first to investigate the routing problem and scheduling problem simultaneously in the ATM domain, and it is found that a time-based separation concept and a flight time weighting scheme applied in the proposed algorithm allows for horizontal separation-compliant routing and scheduling for each flight. Simulation results show that the current flight planning approach would benefit from the proposed R&S algorithm that provides detailed flight plans in a less computation time. Another part of this thesis focuses on the extension of the R&S algorithm to deal with multiple heterogeneous aircraft arriving at multiple airports, and also to cope with three-dimensional route network. With these extensions, the proposed R&S algorithm can be adopted to handle a wider range of operational conditions represented by various combinations of aircraft types in a fleet and neighbour-dependent separation requirements. Numerical simulation using a simple route network model shows that the R&S algorithm can find

the near-optimal route and schedule within polynomial time. As a more realistic case study, we tested the algorithm into the London Terminal Manoeuvring Area (LTMA). The numerical experiment shows that the algorithm provides a separation-compliant route and schedule for multiple heterogeneous aircraft in the three-dimensional LTMA efficiently.

By modifying the proposed algorithm, we address flight planning problems that arise in drone delivery, which is one of the most promising applications of the UTM system. As a preliminary study, we demonstrate two last-mile delivery cases (1-to-$\mathcal{M}$) and one first-mile delivery case ($\mathcal{M}$-to-1) within a route network over roads. The results of each case show that detailed flight plans could support analysis of the route network capacity and help to establish requirements for safe and efficient operations. On the basis of this observation, the analysis of the structured urban airspace capacity is performed for four different types of drone delivery operation (1-to-$\mathcal{M}$, $\mathcal{M}$-to-1, $\mathcal{N}$-to-$\mathcal{M}$, and $\mathcal{M}$-to-$\mathcal{N}$) using the proposed algorithms, where we suggest four intuitive metrics calculated from the detailed flight plans. We apply two different sequencing algorithms (First Come First Served algorithm and Last Come First Served algorithm) - an outer loop of the R&S algorithms - for each operation type. Monte Carlo simulation results suggest to use either more efficient sequencing algorithm or both of the algorithms together in a timely manner for each operation type. From the simulation results, we could expect that the proposed algorithms provide the analysis and suggestions for designing urban airspace to support designers, regulators, and policymakers.

Collectively, the algorithms proposed in this thesis may play a key role in many network-based transport planning problems regarding effective and safe operations, along with future works on extension of the algorithm to real-time planning algorithms and to other transportation systems.

To my parents, and my sister . . .
for their unconditional supports

Contents

List of Figures

List of Tables

Acknowledgment

Firstly, I would like to express my sincere gratitude to my supervisors Dr. Hyo-Sang Shin and Prof. Antonios Tsourdos for invaluable support, patience and guidance throughout my Ph.D at Cranfield university.

I am also extremely thankful to my friends, thank you for listening, offering, and supporting me through this entire process.

Last, but far from least, I want to express my deepest gratitude to my parents and sister who trust and support me as always.

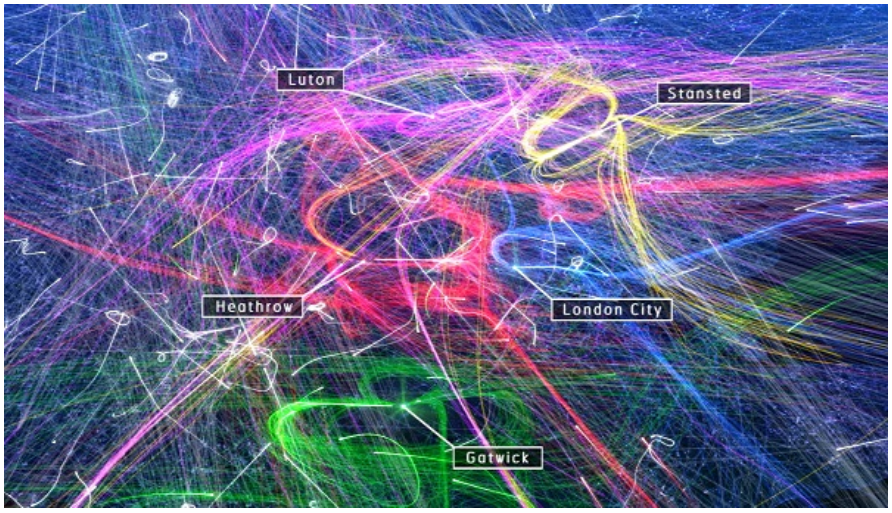
Chapter 1

Introduction

1.1 Background and Motivation

Providing a route (*routing*) and generating a schedule (*scheduling*) for each vehicle plays an important role in the effective and safe operations of many transportations systems [1–4]. Due to the complexity of solving both the routing problem and scheduling problem together at the same time, however, most of the literature have addressed the problems separately [1–3, 5–9].

Especially, in the Air Traffic Management (ATM) system which is one of the representative and complex transportation systems, Air Traffic Controllers (ATCOs) have responsibilities of enforcing routing (in practice known as *path control*) and scheduling (in practice known as *speed control*) for safe and efficient operations based on the initial flight plans that include routes and schedules without considering separation. However, because the flight plans can not manage separation, the initial flight plans have influences only to a very limited extend on ATCOs' decision on strategic and tactical phases. Advances in navigation technology and steadily increasing demand for air transportation will require more detailed flight plans, and the role of the flight plans, which could be automated, will become more important to meet the demand rather than hiring more ATCOs. This is because the complexity of the routing and scheduling problem increases exponentially as the number of flights increases [10].



(a) The real traffic in the London Terminal Manoeuvring Area (LTMA) including five airports (Luton airport, Stansted airport, Heathrow airport, London city airport, and Gatwick airport)

(b) NASA's concept for a possible UTM system

Figure 1.1: Examples of the need of routing and scheduling[1]

[1]Images downloaded in February 2018 from (a) `https://nats.aero/blog/2015/07/we-cant-wait-for-another-runway-before-acting-on-capacity/`, (b) `https://utm.arc.nasa.gov/index.shtml`, respectively.

Also, there is a need for routing and scheduling in an Unmanned Aircraft System (UAS) Traffic Management (UTM) system for Very Low Level (VLL) urban airspace, which is believed to bring many beneficial civilian applications from goods delivery and agricultural monitoring to search and rescue [11]. As there is no established infrastructure to manage UAS operations safely, concept studies for the UTM system are underway [12–15]. One of them is the study of airspace structure designs and its capacity analysis, in which the flight planning is required for more specific and detailed analysis [9]. However, it is shown that there can be an conservative capacity analysis of the airspace structure due to the existing approaches cannot take into account practical factors, and solve the spatial problem and temporal problem separately.

1.2 Research Aim and Objectives

Research Aim

This research aims to develop practical and expandable algorithms that solve both the routing and scheduling problems simultaneously to support effective and safe multiple agent operations in transportation systems. These algorithms should satisfy a variety of operational constraints and ensure safe operation for a specific transportation system whilst minimising a performance index.

Objectives

Amongst many transportation systems, this research mainly focuses on following three applications: multiple aircraft flight planning in a Terminal Manoeuvring Area (TMA); multiple small UAS routing and scheduling in VLL urban airspace for drone delivery services; and VLL urban airspace capacity analysis for the drone delivery services. Specific objectives (O1∼O4, shown in Figure 1.2) of this research are as follows:

O1. Development of a routing and scheduling algorithm for multiple aircraft arriving at a single airport (Chapter 2): The first objective of this dissertation is

to develop an algorithm for generation of horizontal separation-compliant routes and speed profiles for multiple homogeneous aircraft arriving at a single airport. Here, a time-based separation concept applicable to merging points will be introduced as an essential part of the algorithm.

O2. Extension to multiple aircraft arriving at multiple airports (Chapter 3): In order to test the routing and scheduling algorithm proposed in Chapter 2 to more general arrival cases, the algorithm is to be extended to allow to consider the followings: multiple heterogeneous aircraft; three-dimensional airspace; multiple airports in a TMA; and neighbour-dependent separation requirements.

O3. Extension to drone delivery applications (Chapter 4): When implementing the algorithms into drone delivery missions, the algorithms should consider the flight routing and scheduling from departure to arrival because the flight time is relatively short. This Chapter will address two drone delivery operation types: from a single departure point to multiple landing points case (1-to-$\mathcal{M}$); and from multiple departure points to a single landing point case ($\mathcal{M}$-to-1).

O4. Implementation of the algorithms into urban airspace capacity analysis (Chapter 5): The algorithms introduced will be extended to deal with more general operation types, and its results will be utilised for analysing the structured urban airspace capacity.

1.3 Contribution to Knowledge

Contributions of the thesis, found by addressing the prescribed objectives, are as follows. Blocks in Figure 1.2 illustrates each chapter's main contribution, and shows how all the parts of the thesis are related to each other. Note that each block was submitted to or published in a peer-reviewed conference or a journal as a paper, and the papers are listed in Section 1.5.

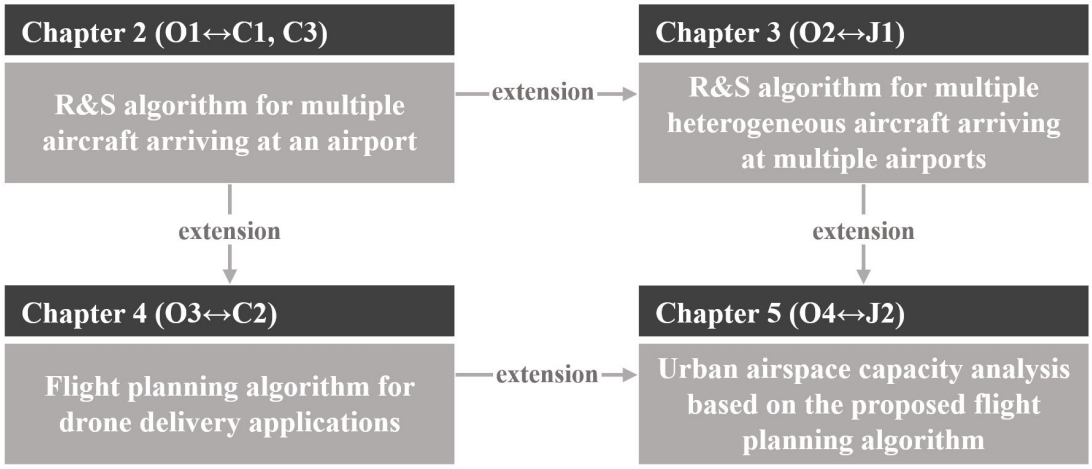


Figure 1.2: The organisation of the thesis (**O#**: Objectives in Section 1.2, and **J# (C#)**: Publications in Section 1.5)

1.3.1 Routing and Scheduling Algorithm for Multiple Homogeneous Aircraft Arriving at an Airport

In a TMA, where tactical changes are required to manage separation between flights, the primary task is to decide the route and schedule of each flight from an initial waypoint to the destination. The routing problem and scheduling problem are usually solved separately because of the separation requirements and operational factors. This study proposes a routing and scheduling algorithm based on graph theory by applying the flight time weights to the graph. Unlike most of the existing approaches handling the routing problem and scheduling problem, the proposed algorithm solves the two problems simultaneously. Therefore, the main contributions of this study are as follows: (1) increasing the solution search space, (2) providing polynomial-time algorithms, (3) and providing the detailed separation-compliant speed profile of each flight. This study demonstrates the advantages of the proposed approach through a case study that considers 23 aircraft arriving at the London Heathrow airport in London TMA. The results show that the proposed algorithm provides a separation-compliant route and schedule for each flight. Also, the results are compared with Regulated Tactical Flight Model (RTFM) obtained from Demand Data Repository 2 (DDR2) [16]. The

details are described in Chapter 2.

1.3.2 Routing and Scheduling Algorithm for Multiple Heterogeneous Aircraft Arriving at Multiple Airports

This study focuses on extending the proposed algorithm to take into account the factors not incorporated in Chapter 2. We first extend the algorithm to deal with routing and scheduling problems of categorised multiple heterogeneous aircraft. Namely, a different feasible speed range for each category can be used, and separation requirements, which vary depending on the leading and following aircraft, can be considered. Second, the extended algorithm can cover routing and scheduling problem in a TMA where there are two or more airports. This can be achieved by resolving sub-problems of finding an airport, which allows more efficient airport operations. For the multiple airports case, we need to take into account crossing points that often found in this case. Namely, vertical separation is introduced as well as horizontal separation for each pair of aircraft. Also, we expand from two-dimensional to three-dimensional airspace. This study demonstrates the expandability of the extended algorithm through a case study that considers multiple heterogeneous aircraft arriving at multiple airports. The results show that the extended algorithm can tackle more operational factors that cannot be covered by the algorithm in Chapter 2.

1.3.3 Flight Planning Algorithm for Drone Delivery Applications

This study focuses on extending the algorithm proposed in Chapter 2 to consider the entire flight phases in the drone delivery application (first-mile delivery and last-mile delivery) that is the most envisioned application of using small Unmanned Aircraft System (sUAS). A primary assumption considered in this study is that the minimum departure separation is predetermined while the departure sequence is not necessarily predetermined for the last-mile delivery. One of the factors determining the efficiency of the last-mile delivery is the departure sequence which is determined by the outer loop

of the algorithm. Here, departure time for each drone is updated to fulfil the minimum departure separation while determining the departure sequence. As a preliminary study, we demonstrate the algorithm through case studies. The results show that the algorithm can deal with the entire flight phases from departure to arrival of the drone delivery applications. Also, the results show that the efficiency of the operations is determined by the outer loop algorithm.

1.3.4 Urban Airspace Capacity Analysis using the Flight Planning Algorithms

A route network-based urban airspace is one of the initial operational concepts of managing the high-density Very Low Level (VLL) urban airspace for Unmanned Aircraft System (UAS) Traffic Management (UTM). For the conceptual design of urban airspace, it is important to provide an appropriate analysis of urban airspace to stakeholders for designing rules and regulations. The aim of this chapter is to apply the proposed approach to analyse the urban airspace capacity, and to provide analysis and suggestions for each operation case to assist airspace designers, regulators, and policymakers. We discuss the urban airspace capacity by applying different flight planning algorithms and comparing its results for different operation types. Four drone delivery operation cases (1-to-$\mathcal{M}$, $\mathcal{M}$-to-1, $\mathcal{N}$-to-$\mathcal{M}$, and $\mathcal{M}$-to-$\mathcal{N}$) are analysed in terms of the following four metrics: total flight time; total flight distance; mission completion time; the number of conflicts. The metrics can be calculated from the simulation results obtained by the proposed flight planning algorithm. The algorithm consists of an inner loop algorithm, which is based on the graph theory to calculate the flight plan of each sUAS, and a changeable outer loop algorithm, which determines arrival and the departure sequences. For each operation case, we apply two different outer loops with the same inner loop to suggest an appropriate sequencing algorithm. Monte Carlo simulation results show tendencies for each type of operation with regard to the outer loop algorithms and the number of sUASs, and we analyse the results and suggest to use either one of the two outer loop algorithms or choose the appropriate one depending on the number of UASs.

1.4 Organisation of the Thesis

The rest of the thesis is organised as follows (also as shown in Figure 1.2). Chapter 2 proposes the routing and scheduling algorithm for multiple aircraft arriving at a single airport based on graph theory. Chapter 3 extends the algorithm to accommodate more general routing and scheduling problems such as arrivals at multiple airports, heterogeneous aircraft, three-dimensional separation. Chapter 4 extends the algorithm to accommodate the routing and scheduling problem considering the entire flight phases of drone delivery. Chapter 5 implements the proposed algorithms for analysing the structured urban airspace capacity for different drone delivery cases, and suggests an efficient sequence allocation algorithm for each case. Lastly, the thesis ends with conclusions and suggestions for future works in Chapter 6. Please note that the thesis is *paper-format*, consisting of each chapter submitted as a self-sufficient individual paper.

1.5 The List of Published/Submitted Works

The following papers were submitted or published in relation to this PhD research.

Journal Papers

J1. **S. Bae**, H.S. Shin, A. Tsourdos, "Network-Based Multiple Aircraft Routing and Scheduling Algorithm Arrival at Multiple Airports," *Journal title* (to be submitted)

J2. **S. Bae**, H.S. Shin, A. Tsourdos, "Structured Urban Airspace Capacity Analysis for Four Drone Delivery Cases," *Journal title* (to be submitted)

Peer-reviewed Conference Papers

C1. **S. Bae**, H.S. Shin, C.H. Lee, A. Tsourdos, "A New Multiple Commercial Aircraft Routing and Scheduling Algorithm for Terminal Manoeuvring Area," in *2018*

IEEE/AIAA 37th Digital Avionics Systems Conference (DASC), London, UK, 23–27 Sep 2018

C2. **S. Bae**, H.S. Shin, A. Tsourdos, "A New Graph-Based Flight Planning Algorithm for Unmanned Aircraft System Traffic Management," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*, London, UK, 23–27 Sep 2018

C3. **S. Bae**, H.S. Shin, A. Tsourdos, "Multi-Step Trajectory Optimization for ATM Based on Approximated Optimal Path," in *30th Congress of the International Council of the Aeronautical Sciences (ICAS 2016)*, Daejeon, Korea, 25–30 Sep 2016

References

[1] H. Balakrishnan and B. Chandran, "Scheduling Aircraft Landings Under Constrained Position Shifting," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*. Reston, Virigina: American Institute of Aeronautics and Astronautics, aug 2006, p. 6320.

[2] D. R. Isaacson, a. V. Sadovsky, and D. Davis, "Tactical Scheduling for Precision Air Traffic Operations: Past Research and Current Problems." *Journal of Aerospace Information Systems*, vol. 11, no. 4, pp. 234–257, 2014.

[3] H. Fazlollahtabar and M. Saidi-Mehrabad, "Methodologies to Optimize Automated Guided Vehicle Scheduling and Routing Problems: A Review Study," *Journal of Intelligent & Robotic Systems*, vol. 77, no. 3-4, pp. 525–545, 2015.

[4] R. Breil, D. Delahaye, L. Lapasset, and E. Feron, "Multi-agent systems for air traffic conflicts resolution by local speed regulation and departure delay," in *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, 2016, pp. 1–10.

[5] Hanbong Lee and H. Balakrishnan, "A Study of Tradeoffs in Scheduling Terminal-Area Operations," *Proceedings of the IEEE*, vol. 96, no. 12, pp. 2081–2095, dec 2008.

[6] A. V. Sadovsky, "Application of the Shortest-Path Problem to Routing Terminal Airspace Air Traffic," *Journal of Aerospace Information Systems*, vol. 11, no. 3, pp. 118–130, mar 2014.

[7] A. V. Sadovsky, D. Davis, and D. R. Isaacson, "Efficient Computation of Separation-Compliant Speed Advisories for Air Traffic Arriving in Terminal Airspace," *Journal of Dynamic Systems, Measurement, and Control*, vol. 136, no. 4, p. 041027, may 2014.

[8] A. Rezaei, A. V. Sadovsky, and J. L. Speyer, "Existence and Determination of Separation-Compliant Speed Control in Terminal Airspace," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 6, pp. 1374–1391, jun 2016.

[9] M. F. B. Mohamed Salleh, C. Wanchao, Z. Wang, S. Huang, D. Y. Tan, T. Huang, and K. H. Low, "Preliminary Concept of Adaptive Urban Airspace Management for Unmanned Aircraft Operations," in *2018 AIAA Information Systems-AIAA Infotech @ Aerospace*, no. January. Reston, Virginia: American Institute of Aeronautics and Astronautics, jan 2018, pp. 1–12.

[10] D. R. Isaacson, a. V. Sadovsky, and D. Davis, "Tactical Scheduling for Precision Air Traffic Operations: Past Research and Current Problems." *Journal of Aerospace Information Systems*, vol. 11, no. 4, pp. 234–257, 2014.

[11] S. Bradford, "Version 1.0 of the Unmanned Aireraft Systems (UAS) Traffic Management (UTM) Concept of Operations," Federal Aviation Administration, Tech. Rep., 2018.

[12] T. Prevot, J. Homola, and J. Mercer, "From Rural to Urban Environments: Human/Systems Simulation Research for Low Altitude UAS Traffic Management (UTM)," in *16th AIAA Aviation Technology, Integration, and Operations Conference*. Reston, Virginia: American Institute of Aeronautics and Astronautics, jun 2016, p. 3291.

[13] Geister and Dagi, "Concept for Urban Airspace Integration DLR U-Space Blueprint," Institute of Flight Guidance, Tech. Rep. December, 2017. [Online]. Available: http://www.dlr.de/fl/desktopdefault.aspx/tabid-11763/20624{_}read-48305/

[14] C. A. CASTIGLIONI, A. S. RABUFFETTI, G. P. Chiarelli, G. BRAMBILLA, and J. Georgi, "Unmanned aerial vehicle (UAV) application to the structural health assessment of large civil engineering structures," in *Fifth International Conference on Remote Sensing and Geoinformation of the Environment (RSCy2017)*, G. Papadavid, D. G. Hadjimitsis, S. Michaelides, V. Ambrosia, K. Themistocleous, and G. Schreier, Eds. SPIE, sep 2017, p. 29.

[15] M. Cannioto, A. D'Alessandro, G. Lo Bosco, S. Scudero, and G. Vitale, "Brief communication: Vehicle routing problem and UAV application in the post-

earthquake scenario," *Natural Hazards and Earth System Sciences*, vol. 17, no. 11, pp. 1939–1946, nov 2017.

[16] Eurocontrol, "Demand Data Repository 2," 2017. [Online]. Available: https://www.eurocontrol.int/articles/ddr2-web-portal

Chapter 2

Routing & Scheduling Algorithm for Multiple Arrivals at an Airport

2.1 Introduction

Since air traffic demand increases continuously, modernisation projects SESAR and NextGen [1, 2], and a considerable amount of research [3–16] for the future Air Traffic Management (ATM) system have been conducted to improve safety and efficiency of flights. In future ATM environment, the operation of air traffic requires more detailed information such as separation-compliant speed profile for each flight than the current ATM system because the flights will follow *aRea NAVigation (RNAV)* and *Required Navigation Performance (RNP)* procedures in the networks [10]. Typically, scheduling means determining an arrival sequence with time slots in the flight planning stage, but the terminology of *scheduling* throughout this study is used for determining a separation-compliant speed profile, and arrival time at each waypoint (i.e., an arrival sequence) in the flight planning stages. Thus, the schedule helps Air Traffic Controllers (ATCOs) manage safe and efficient operations in the tactical planning stage. The terminology of *routing* for this study is for determining a flight route composed of a set of linear segments (nominal routes).

The requirements of the future ATM system lead to an entire class of routing and

scheduling problems, which depend on airspace configurations, aircraft performance constraints, and operational constraints. For example, a Terminal Manoeuvring Area (TMA) is a dense and complex route network compared to an en-route area, which requires heavy workload of ATCOs in charge of routing and scheduling for each flight in their control area and assuring separation between each pair of aircraft. ATCOs resolve these problems based on their experience, intuition and some scheduling rules without using formally defined performance indices [16]. Without mathematical formulations or quantitative metrics, however, it is difficult to obtain meaningful results in post-analyses to improve the airspace efficiency and safety, although the system generally works well.

A great deal of previous research into the issue has focused on resolving the routing and scheduling problem using optimisation techniques. Determining a flight route consisting of a finite set of waypoints in the airspace network is often modelled as a directed graph, and a schedule as a chart that optimises the estimated time of arrival for each flight [3–8]. However, such approaches only consider the flight schedule at a runway, not entire speed profiles. One limitation of the model is that the scheduling model does not take into account the minimum separation during the flight, which is one of the most important operational requirements for safety without adding artificial constraints. In [6, 17], authors have attempted to model flight scheduling problems using a Mixed-Integer Linear Programming (MILP) and a Dynamic Programming (DP) to maximise runway throughput while enforcing minimum separation in time. However, the drawback of such approaches is that it is computationally highly demanding [18, 19], although these are scalable in the number of flights. More recently, in [9, 11–15], authors have attempted to address routing and scheduling problems separately or sequentially. These approaches have advantages such as fast computational time, proposing a mathematical model, proving the existence, and determination of separation-compliant speed. However, we expect that if two problems are solved separately or sequentially, the scheduling solution found satisfying separation may degrade the optimality of routing results.

Our main idea is that if the Estimated Time of Arrival (ETA) at each waypoint for each flight can be shared with each other, routing and scheduling solutions satis-

fying the minimum separation can be formulated using a weighted directed graph and may show computational performance suitable for real-time applications in the flight planning stage. Also, we would obtain a flight route and a schedule for each flight simultaneously by solving the shortest path problem of the airspace graph by means of any shortest path algorithms. Thus, we could find solutions in a larger solution space than the existing sequential approaches.

Firstly, in this study, we propose a novel algorithm based on the weighted directed graph to solve a routing and scheduling problem for a single flight to be planned. Data of planned flights are utilised to calculate weights expressed in flight time for each edge of the airspace graph to satisfy the time-based separation requirement in which the objective function of the weights is to minimise the total flight time. Namely, a flight route found using the shortest path algorithm includes a speed profile that minimises their flight time. The speed profile sought here is a separation-compliant speed advisory for ATCOs as the speed profiles assure the minimum separation at all times. Furthermore, we can obtain the multiple flights' routes and speed profiles by iteratively solving the routing and scheduling problem for each flight with the First Come First Served (FCFS) algorithm to determine an arrival sequence.

We expect the following merits of the proposed algorithm; 1) by simultaneously addressing the routing and scheduling problem while satisfying the minimum separation requirement, the algorithm could mitigate the disadvantage of sequential approaches (i.e., latter stage (scheduling and/or separation) may degrade the optimality of the former results (routing and/or scheduling)); 2) as the proposed algorithm can significantly improve computational performance compared to the previous research, we highly expect that it can be utilised as an automated real-time Decision Support Tool (DST) that helps ATCOs' duties; and 3) it is capable of capturing realistic aspects of the ATM system such as time-based separation, speed restrictions, nominal segments and waypoints, etc.

The rest of this study is organised as follows. Section 2.2 describes the problem formulation for a single flight routing and scheduling. Section 2.3 propose the algorithm for multiple flights routing and scheduling. A case study of 23 flights arriving at London Heathrow airport in the London TMA (LTMA) is given in Section 2.4. Finally,

concluding remarks follow in Section 2.5.

2.2 Problem Statement for a Single Flight Routing and Scheduling

This section consists of five subsections: (a) a description of airspace networks for operational purposes; (b) a description of a time-based separation concept; (c) assumptions considered in this study; (d) mathematical modelling for the routing and scheduling problem; (e) and a description of the flight time based weighting scheme.

2.2.1 Airspace Network

For ATM operational purposes, airspace is often considered as a region with a set of waypoints. Some pairs of waypoints are connected, and these are called (route) segments. STandard Arrival Routes (STARs) provide waypoints and flight routes consisting of a set of some waypoints, flight level limitation, heading angle, etc., that airspace users should follow such as altitude constraints, track angle as shown in Figure 2.1. For the routing problem, we make use of the segments published in STARs instead of using new potential segments such as those proposed in [20].

2.2.2 Time-Based Separation Concept

Conflict detection is activated when separation (time-based separation or distance-based separation) of two flights is less than a minimum separation criterion. In this study, we utilise the time-based separation that is determined by the Wake Turbulence Category (WTC) of the leading flight and the following flight as shown in Table 2.1 instead of the widely used distance-based separation (e.g., 5 NM prescribed by ICAO Doc 4444) to stabilise the time spacing between arrival pairs of aircraft across headwind conditions to recover the lost landing rate currently experienced [21, 22]. By satisfying the separation assurance adjusting the Estimated Time to Arrival (ETA), we define it as

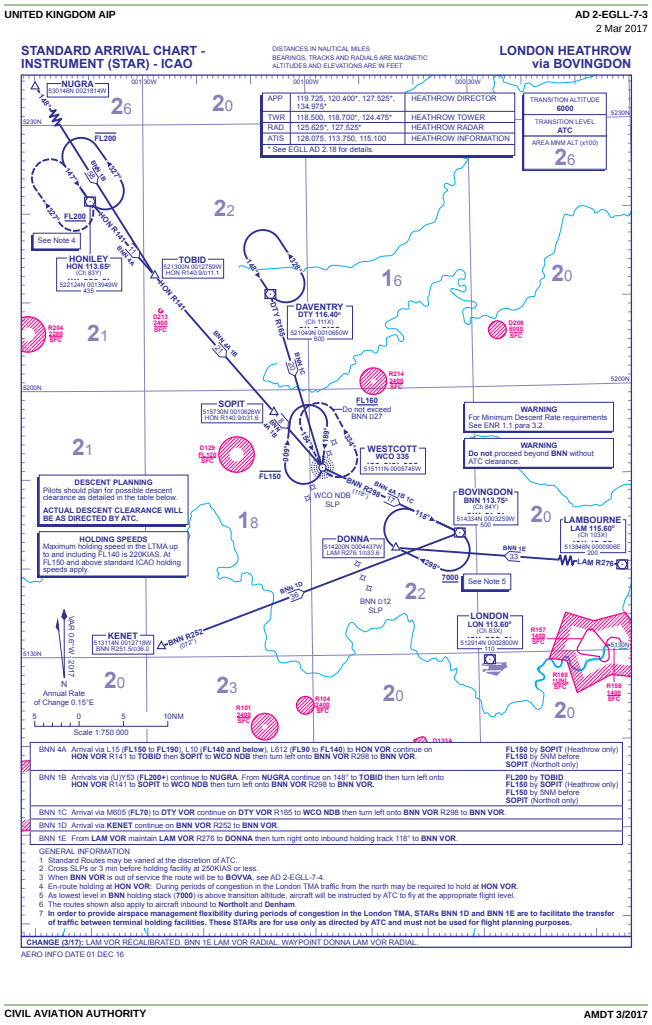


Figure 2.1: A London Heathrow STAR (AD 2-EGLL-7-3)

the Separation Assured Estimated Time of Arrival (SAETA), at each waypoint rather than adjusting the speed on the routes, we obtain results that satisfy the minimum separation at each waypoint as well as the separation at each route segment. The concept allows the time-based separation to be satisfied at merging points as shown in Figure 2.2. In the routing and scheduling problem, our main purpose is to minimise flight time while satisfying the separation requirement, where it is necessary to contain SAETA of each flight at each waypoint. Here we assume that the speed profile is a set of constant

values, where the form of the speed profile is the same as one in flight plans.

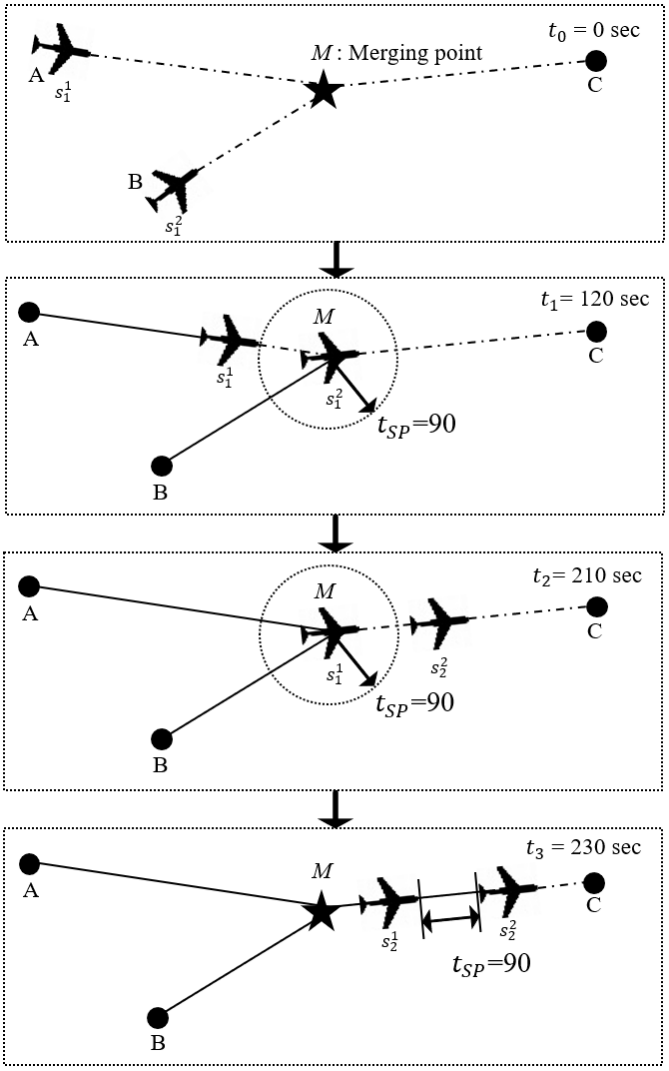


Figure 2.2: Time-based separation concept for merging points (homogeneous flight 1, α_1, and flight 2, α_2, fly from its origin A and B to the same destination C via a merging point M at the speed $\{s_1^1, s_2^1\}$ and $\{s_1^2, s_2^2\}$, respectively. The superscript and subscript of s are flight index and segment index, respectively)

At the time t_0 in Figure 2.2, flight α_1 and α_2 fly toward the waypoint C through the same merging point, *M*, at the speeds s_1^1 and s_1^2, respectively. The time t_1 when α_2 just passes through *M* at the speed of s_1^2 is stored at the merging point $T(M)$, and no flight can pass through this point for t_{SP} seconds before and after the SAETA t_1. The

SAETA t_2 is stored in $T(M)$ in the same way. Then, α_1 and α_2 traverse toward C at the speeds of s_2^1 and s_2^2, respectively. If α_1 and α_2 fly from *M* to C at the speeds of s_2^1 and s_2^2, respectively, the separation between two flights will always satisfy the minimum separation requirement or greater than that on the segment between *M* and C.

$$T(A) = \{0s\}$$

$$T(B) = \{0s\}$$

$$T(M) = \{0s, 120s, 210s\}$$

$$T(C) = \emptyset$$

The generic reference time-based separation depending on the following and leading flights is used for setting up t_{SP} as shown in Table 2.1. Obtained time data, $T(M)$, at each waypoint is used to formulate Equations (2.4)-(2.9) that will be described in Section.2.3.

Table 2.1: Generic reference time-based separations [23]

Leader \ Follower	**A380**	**Heavy**	**Medium**	**Light**
A380	60s	145s	167s	189s
Heavy	60s	98s	122s	145s
Medium	60s	60s	60s	122s
Light	60s	60s	60s	60s

2.2.3 Assumptions

The following assumptions are made for this study: (a) each flight route is represented by a series of linear segments as shown in Figure 2.3; (b) the speed at each segment is a constant value; (c) aircraft speed applied in this study is True AirSpeed (TAS); (d)

arrival sequence is undetermined; and (e) uncertainties produced by external sources are neglected.

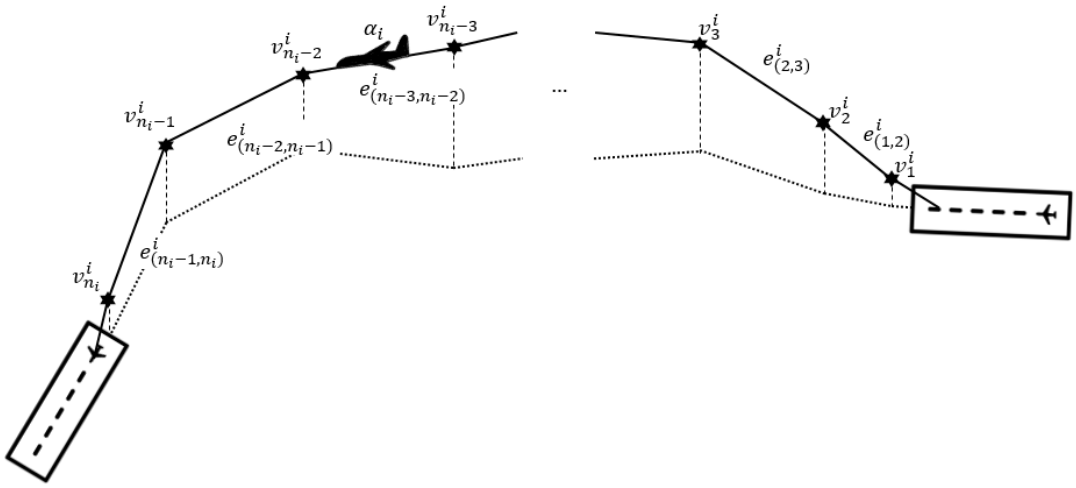


Figure 2.3: A series of linear segments (i.e., the flight route of α_i from its initial waypoint v_1^i to destination $v_{n_i}^i$)

2.2.4 Mathematical Modelling

Our focus is on routing and scheduling a finite set

$$\mathcal{A}^+ = \{\alpha_1, \alpha_2, \ldots, \alpha_i\} \tag{2.1}$$

of i flights in a TMA, each flight $\alpha_i \in \mathcal{A}^+$ to fly from its initial waypoint to its destination (runway), both specified as an input to the problem within feasible speed ranges and a route network. Note that, α_i refers to the flight in this study. The route network in the TMA is modelled as a directed graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$, called airspace graph. In the airspace graph, each vertex $v \in \mathcal{V}$ is a waypoint candidate to be traversed through Euclidean space of dimension two or three. Each edge $e \in \mathcal{E}$, corresponding a rectifiable curve, is a segment between some pair of waypoints in the airspace graph. In this model, a feasible flight route of α_i in the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$ is defined as follows:

Definition 1 As given in the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$, a flight α_i to be routed in $\mathcal{G}$, and the initial waypoint $v_1^i \in \mathcal{V}$ and the final waypoint $v_{n_i}^i \in \mathcal{V}$, a flight route denoted by $p(\alpha_i)$ in $\mathcal{G} = (\mathcal{E}, \mathcal{V})$ is defined by a sequence of waypoints.

In the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$, therefore, there can be an abundance of flight route candidates denoted by $\mathcal{C}$ that satisfy the conditions as described in Definition 1. Through Definition 1, flight route candidate $p(\alpha_i) \in \mathcal{C}$ can be given a corresponding flight route (a set of waypoints) as follows:

$$p(\alpha_i) : v_1^i, v_2^i, \ldots, v_{n_i}^i.$$

For each flight route $p(\alpha_i), \forall \alpha_i \in \mathcal{A}^+$, there is a set of edges connecting waypoints through this flight route, which is as follows:

$$\mathcal{E}(p(\alpha_i)) : e_{(1,2)}^i, e_{(2,3)}^i, \ldots, e_{(n_i-1,n_i)}^i.$$

Wherein through this formulation, for each flight route candidate, it is assumed that a performance index for each flight route $\mathcal{E}(p(\alpha_i))$ can be quantified as a set of positive numeric weighting values such as distance, fuel burnt, as follows:

$$\mathcal{W}(p(\alpha_i)) : w_1^i, w_2^i, \ldots, w_{n^i-1}^i$$

Then, the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$ is transformed into a weighted directed graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W})$ by assigning a weight to each segment $e \in \mathcal{E}$. In the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W})$, each flight route can be obtained by summing all weights in $\mathcal{W}(p(\alpha_i))$, as follows:

$$\mathcal{T}(p(\alpha_i)) = \sum_{j=1}^{n_i-1} w_j^i \tag{2.2}$$

Based on the airspace graph, $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W})$, the routing problem that minimises a performance index can be defined as follows:

Definition 2 Given an airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W})$ and all corresponding flight route candidates $\mathcal{C}$, the routing problem is defined as finding a flight route (or a sequence of

waypoints) such that

$$p^*(\alpha_i) = \underset{p(\alpha_i)\in\mathcal{C}}{\operatorname{argmin}} \mathcal{T}(p(\alpha_i)) \tag{2.3}$$

The optimal flight route $p^*(\alpha_i)$ can be found by using the well-known shortest path algorithms such as Dijkstra's algorithm or the exhaustive search algorithm [18]. Although the optimal flight route $p^*(\alpha_i)$ can be obtained according to Definition 2, the optimality of the flight route might be disturbed in the following scheduling and conflict resolution stages to satisfy the minimum separation. Such a sequential approach can cause not only degrading optimality but also high levels of workload to ATCOs.

Our main idea for achieving the objective, which is to maintain runway throughput by minimising each flight time, is to assign a flight time weight to each segment (edge) of the airspace graph. Then, each flight schedule can be obtained by finding the optimal flight route in the airspace graph of which weights are the flight time. In this study, we set a flight time of the flight as weight $w \in \mathcal{W}$ to segment $e \in \mathcal{E}$ of the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W})$. Note that, the flight time and the speed of aircraft are mutually interchangeable using geographic data of the airspace graph obtained from STARs.

Another issue we have pursued is to satisfy the minimum separation requirement between each pair of aircraft. To assign a flight time that satisfies the minimum separation to each edge of $\mathcal{G}$, we need time data containing the SAETA for each waypoint discussed in Section.2.2.2. Time data, T, will be included in the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W}, T)$ and be used to calculate weights of the airspace graph. By finding a solution of the airspace graph $\mathcal{G}$, then, we can obtain the optimal flight route and its schedule while simultaneously satisfying the minimum separation. Each flight $\alpha_i \in \mathcal{A}^+$ might have different weights $\mathcal{W}_{i,t}$ because of the different initial point, and different specifications of each flight. Thus, flight α_i might have its unique airspace graph $\mathcal{G}_i$ as shown in Figure 2.4. In Section 2.2.5, we describe the flight time weighting scheme to be applied to airspace graphs $\mathcal{G}_i$.

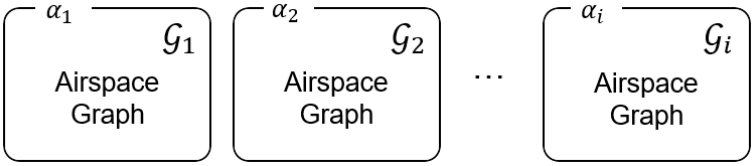


Figure 2.4: Airspace graphs for multiple flights

2.2.5 Calculating Flight Time Weights on $\mathcal{G}_i$ Suitable for Separation-compliant Speed Profiles

This section describes a separation assured flight time weighting scheme for the airspace graph. The weights can be easily converted to separation-compliant speed profiles to support the ATCOs' decision making. A Linear Programming (LP) problem for calculating the weights is then defined follows:

$$\min \quad \sum_{j=1}^{n_i-1} \frac{d_j^i}{s_j^i} \tag{2.4}$$

$$\text{s.t.} \quad \frac{d_1^i}{s_1^i} + \frac{d_2^i}{s_2^i} + \cdots + \frac{d_{n_i-1}^i}{s_{n_i-1}^i} \geq \max T(v_{n_i}^i) + t_{SP} \tag{2.5}$$

$$\frac{d_1^i}{s_1^i} + \frac{d_2^i}{s_2^i} + \cdots + \frac{d_{n_i-2}^i}{s_{n_i-2}^i} \geq \max T(v_{n_i-1}^i) + t_{SP} \tag{2.6}$$

$$\vdots$$

$$\frac{d_1^i}{s_1^i} + \frac{d_2^i}{s_2^i} \geq \max T(v_3^i) + t_{SP} \tag{2.7}$$

$$\frac{d_1^i}{s_1^i} \geq \max T(v_2^i) + t_{SP} \tag{2.8}$$

$$s_{\text{min}}^i \leq s_1^i, s_2^i, \cdots, s_{n_i-1}^i \leq s_{\text{max}}^i \tag{2.9}$$

where s_{min}^i, s_{max}^i, and t_{SP} are the minimum speed, the maximum speed of flight α_i, and the minimum separation time between flights at waypoints as shown in Figure 2.2.

Equation (2.4) of the LP problem, which is the objective function, is to minimise the flight time. Equations (2.5)-(2.8) are for satisfying the minimum separation at each waypoint. As shown in Figure 2.3, for example, if $\mathcal{E}(p(\alpha_i))$ consists of $n_i - 1$ segments for flight α_i, each segment requires a flight time that satisfies the time-based separation at each waypoint. Namely, the number of constraints for the separation is equal to the number of segments. The left hand side of the constraints is the total flight time to the end point of the final segment of each constraint, which must be greater than and equal to the time that satisfies the minimum separation.

For the constraints, segment distance d between waypoints, and the time data T stored in $\mathcal{G}_i$ are required. A flight distance $d(e)$ for segment $e \in \mathcal{E}$ in STARs is calculated by using the Harversine formula [24]. The time data $T(v^i_{n_i})$, which is the SAETA that α_i fly over a waypoint $v^i_{n_i}$. The SAETA is stored in T for waypoint $v \in \mathcal{V}$, and T is updated every time a flight is routed and scheduled, and the updated T is shared with $\alpha_i \in \mathcal{A}^+$. In this study, we use a Sequential Quadratic Programming (SQP) method to solve the LP problem.

Decision variables, $s^i_1, s^i_2, \cdots, s^i_{n_i-1}$, are constant aircraft speeds for a set of segments. The aircraft speeds can be converted into the flight time weights using the following equation:

$$\text{flight time} = \mathcal{W}_{i,t} = \frac{d(\alpha_i)}{s(\alpha_i)} \tag{2.10}$$

where $s(\alpha_i)$ and $d(\alpha_i)$ are a set of constant speeds and a set of flight distances, respectively.

$$s(\alpha_i) : s^i_1, s^i_2, \cdots, s^i_{n_i-1}$$
$$d(\alpha_i) : d^i_1, d^i_2, \cdots, d^i_{n_i-1}$$

The flight times are assigned into weights $\mathcal{W}_{i,t}$. Therefore, a solution of the airspace graph $\mathcal{G}_i = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}, T)$ can simultaneously provide the optimal flight route and its speed profile satisfying the minimum separation requirement at all times. The permissible speed range constraint in Equation (2.9) for each flight is only a function of the International Civil Aviation Organization (ICAO) aircraft approach category [25, 26].

We also construct flight distance weights $\mathcal{W}_d$ on $\mathcal{G}_i$ to reflect the flight distance as a second criterion. The flight distance weights are necessary to prioritise for the multiple flights routing and scheduling problem when two or more aircraft arrive at the runway at the same time, more details in Section.2.3. In summary, the problem that motivated this study can be formulated as follows:

Problem 1 Given an airspace graph $\mathcal{G}_i = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}, \mathcal{W}_d, T)$, flight α_i to be routed and scheduled in $\mathcal{G}_i$, and its initial waypoint $v_1^i \in \mathcal{V}$ and its destination $v_{n_i}^i \in \mathcal{V}$, reachable from the initial waypoint, of each flight $\alpha_i \in \mathcal{A}^+$, construct a flight route $p(\alpha_i)$ and a speed profile $s(\alpha_i)$ in $\mathcal{G}_i$ such that

- the separation requirement is satisfied from planned flights,
- the speed profile of each flight must be within its feasible speed range,
- the airspace graphs are updated every time when flight $\alpha_i \in \mathcal{A}^+$ finds its plan.

In this study, we solve the Problem 1 using Dijkstra's algorithm [27].

2.3 An Algorithm for Solving Routing and Scheduling Problems of Multiple Aircraft Arriving at an Airport

Scheduling of arrival flights can be divided into three stages: initial sequencing stage, modifying schedule stage, and freezing stage [28]. Generally, for the Aircraft Landing Problem (ALP), the initial sequencing stage is based on the FCFS algorithm, which is the landing order that would result, if each flight proceeded to the runway and landed without due consideration of other flights. This approach, however, causes many modifications and an increased workload to ATCOs in the next modifying schedule stage. In this section, an algorithm considering separation between every pair of aircraft is proposed using the FCFS algorithm to reduce the workload of ATCOs by reducing the difference between the stages.

We propose Algorithm 1 to solve the multiple flights routing and scheduling problem while satisfying the minimum separation requirement by iteratively solving Problem 1 with the FCFS algorithm. The proposed algorithm proceeds as follows: (Line 4) given an instance of the problem, our approach first computes a separation-compliant flight route and its schedule for each flight in a set of flights to be planned $\mathcal{A}^+$ using Dijkstra's algorithm (in this stage of every iteration, the meaning of separation means the minimum separation requirement from the planned flights $\mathcal{A}^-$); (Line 5) the FCFS algorithm is used to find a flight that can maximise the runway throughput; (Line 7-8) if the performance index which is the runway arrival time of two or more flight are the same, the second criterion, which is flight distance, is applied to determine the flight to arrive first; (Line 10-11 or Line 14-15) the planned flight is removed from $\mathcal{A}^+$ and stored in a set of planned flight $\mathcal{A}^-$, and the flight route and schedule of the planned flight are shared for flights in $\mathcal{A}^+$; (Line 18) based on the shared data, our approach updates Problem 1 for unplanned flights $\mathcal{A}^+$. Once the inputs of the algorithm are given, the algorithm runs the routing and scheduling process of the multiple flights in the TMA until $\mathcal{A}^+$ is empty and, therefore, $\mathcal{A}^-$ is full.

We illustrate an example in Figure 2.5 for a better understanding of the iteration concept of the algorithm. In the first iteration, each flight $\alpha_i \in \mathcal{A}^+ = \{\alpha_1, \alpha_2, \cdots, \alpha_A\}$ has its airspace graph, and flight α_2 is planned as the first flight to arrive the airport. Then, α_2 is transferred from $\mathcal{A}^+$ to $\mathcal{A}^-$. In the second iteration, the airspace graph of each flight $\alpha_i \in \mathcal{A}^+$ is updated, and flight α_A is planned as the second flight to arrive the airport. Then, α_A is transferred from $\mathcal{A}^+$ to $\mathcal{A}^-$. In the $(A-1)^{\text{th}}$ iteration, flight α_3 is planned, then flight α_1 is planned in the same way.

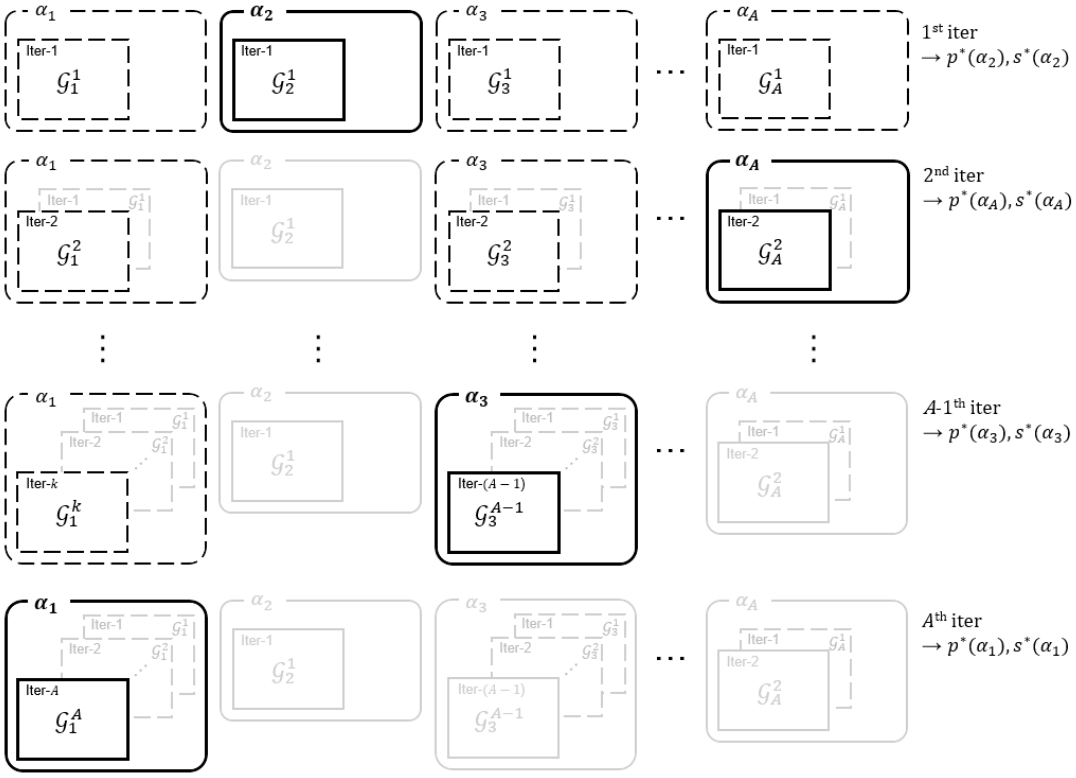


Figure 2.5: Iteratively generated airspace graphs for multiple flights in Algorithm 1

Algorithm 1: Multiple flights routing and scheduling algorithm in a TMA

Input: $v_1^i, v_{n_i}^i \in \mathcal{V}$, $\forall \alpha_i \in \mathcal{A}^+$, airspace information

Output: $p(\alpha_i), s(\alpha_i)$, $\forall \alpha_i \in \mathcal{A}^-$

$k = 1$

generate $\mathcal{G}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}^k, \mathcal{W}_d, T^k)$, $\forall \alpha_i \in \mathcal{A}^+$

while $\mathcal{A}^+ \neq \emptyset$ **do**

  find $p^*(\alpha_i)$ of $\mathcal{G}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}^k, T^k)$, $\forall \alpha_i \in \mathcal{A}^+$ (using Dijkstra's algorithm) ;

  ${\alpha_i}^* \leftarrow \underset{\alpha_i \in \mathcal{A}^+}{\operatorname{argmin}} p^*(\alpha_i)$(using the FCFS algorithm);

  if *There are more than two* $p^*(\alpha_i)$ *exist* **then**

    find $p^*(\alpha_i)$ of $\mathcal{G}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_d, T^k)$ amongst them (using Dijkstra's algorithm);

    $\alpha_i^\dagger \leftarrow \underset{\forall \alpha_i^*}{\operatorname{argmin}} p^*(\alpha_i)$ (using the FCFS algorithm);

    allocate $\alpha_i^\dagger$;

    remove $\alpha_i^\dagger$ from $\mathcal{A}^+$ and store $\alpha_i^\dagger$ in $\mathcal{A}^-$ with $p^*(\alpha_i)$ and $s(p^*(\alpha_i))$;

    share $\mathcal{A}^-$ with $\forall \alpha_i \in \mathcal{A}^+$;

  else

    allocate α_i^*;

    remove α_i^* from $\mathcal{A}^+$ and store α_i^* in $\mathcal{A}^-$ with $p^*(\alpha_i)$ and $s(p^*(\alpha_i))$;

    share $\mathcal{A}^-$ with $\forall \alpha_i \in \mathcal{A}^+$;

  end

  $k = k + 1$;

  update $\mathcal{G}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}^k, \mathcal{W}_d, T^k)$, $\forall \alpha_i \in \mathcal{A}^+$;

end

2.4 Case Study: 23 Arrivals at Heathrow Airport in the LTMA

To demonstrate the proposed algorithm, we conduct a case study of multiple flights routing and scheduling in the London TMA (LTMA). As a preliminary case study, we make the following operational assumptions: (1) only London Heathrow Airport (LHR) of the five airports in the LTMA is considered; (2) only medium type of aircraft is considered; (3)the aircraft are ICAO aircraft approach category C where we increase 10 knots of it because the aircraft speed is higher than the category in real world flights; (4) segments from the holding stacks to LHR are straight lines (namely, there are no holding manoeuvres); and (5) single runway is considered. Results will show the efficiency of our approach and the possibility as ATCOs' DST by providing a flight route and its schedule for each flight, and the optimal trajectory based on the results shows the feasibility of the route and schedule obtained from the proposed algorithm. Also, the results will be converted to the number of flights that can arrive at LHR by the medium type of aircraft per hour, and compared with Regulated Tactical Flight Model (RTFM) obtained from EUROCONTROL DDR2 [29].

London Heathrow (EGLL) STARs in the United Kingdom Aeronautical Information Publication (AIP) [30] are applied to construct an airspace graph, as the flights obtained using the proposed algorithm only follow the nominal route segments. We construct the airspace graph as shown in Figure 2.6, where we simplify the routes from the four holding stacks (BNN, LAM, OCK, BIG) including the Initial Approach Fix (IAF) to the Final Approach Fix (FAF) because of the following practical issues: most of the arrival flights to LHR in peak time holds at the four holding stacks until pilots receive an ATC clearance to maintain the runway throughput.

To construct a realistic scenario, we deal with a routing and scheduling problem of 23 medium aircraft according to the ICAO WTC. More details are given in the Table 2.2. There are 23 aircraft ($\alpha_1 \sim \alpha_{23}$) on the airspace graph as shown in Figure 2.7. Each aircraft should arrive London Heathrow Airport via one of the holding stacks. Table 2.3 and Table 2.4 show the results, also it is graphically depicted in Figure 2.7

Table 2.2: Airspace graph and aircraft information

Airspace graph information	
Airport	London Heathrow (LHR)
STARs	EGLL
# of runways	1
# of fixes (waypoints)	45 (STARs)
# of routes	53 (including 4 holding tracks)
Aircraft information	
# of aircraft	23 (medium aircraft type)
Minimum speed (TAS)	150 knot
Maximum speed (TAS)	250 knot

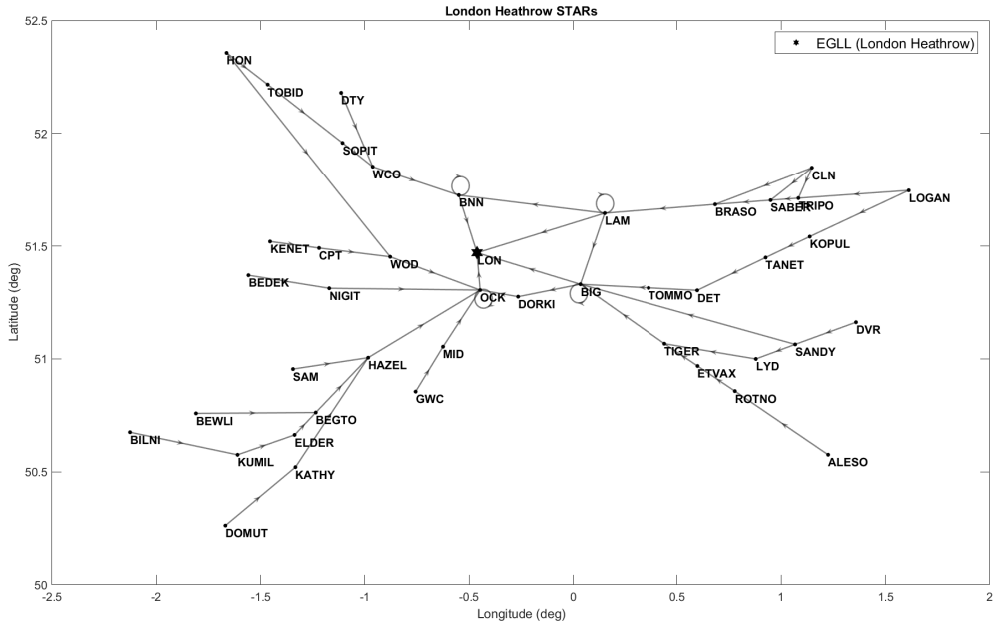


Figure 2.6: The airspace graph for landing to London Heathrow airport (the airspace graph is based on EGLL STARs)

$\sim$ Figure 2.10. Table 2.3 shows the arrival sequence, start waypoint, and the arrival time at the airport. The flight route from the start waypoint to the end waypoint of

each aircraft and its separation-compliant speed profile are shown in Table 2.4, which would help ATCOs' decision making by reducing the gap between each stage. Also, in order to validate whether the obtained flight routes and schedules are feasible or not, we numerically optimise the trajectory based on the results and the Base of Aircraft Data (BADA) v3 performance data [26]. Figure 2.11 shows the optimal trajectory of α_3 with its speed profile shown in Figure 2.12, in which both deviate from the obtained flight route and speed profile due to the proposed algorithm only provide the set of linear segments and the constant speed profile for each flight. The optimal trajectory constrained by the flight route and schedule shows that the proposed algorithm could provide feasible flight routes and schedules for ATCOs' decision making. Please refer to Appendix A for more detail about the trajectory optimisation.

For quantitative comparison relying on the assumptions we made, RTFM obtained from EUROCONTROL DDR2 "21:00 $\sim$ 21:20 Thursday 14 September 2017", is used, as shown in Table 2.5 [29]. In this data, 15 flights landed in 20 minutes, on the other hand, in the proposed algorithm, 23 flights could arrive in 23 minutes. Namely, the proposed algorithm could improve the runway throughput by 23% compared to RTFM at London Heathrow Airport, and might become as a new DST that enables a reduction of the ATCOs' workload. The computation time for calculating the case study is 10.65 seconds on a Windows 10 OS 3.4 GHz desktop computer with 16 GB RAM.

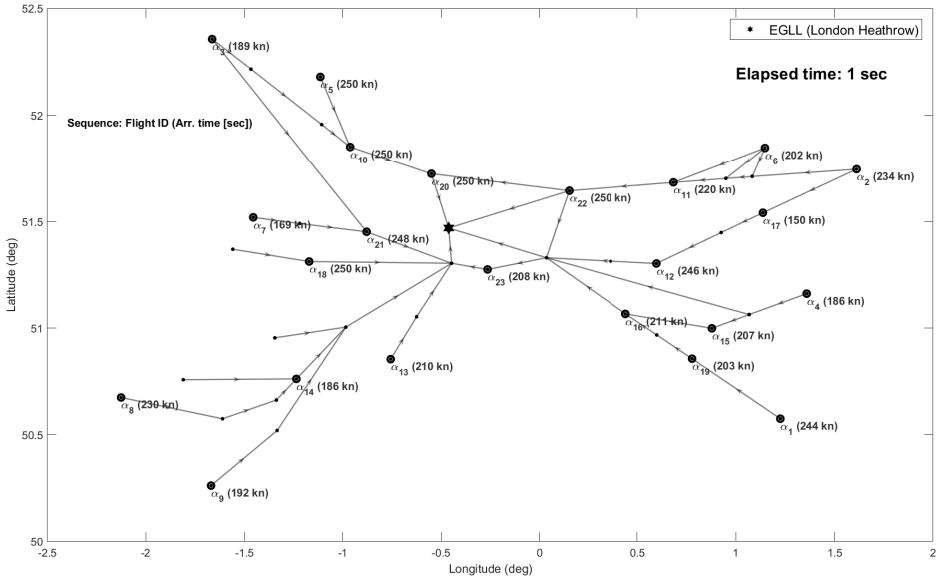


Figure 2.7: Case study results of 23 aircraft arriving at Heathrow airport in the LTMA: positions of aircraft at $t = 1$ sec

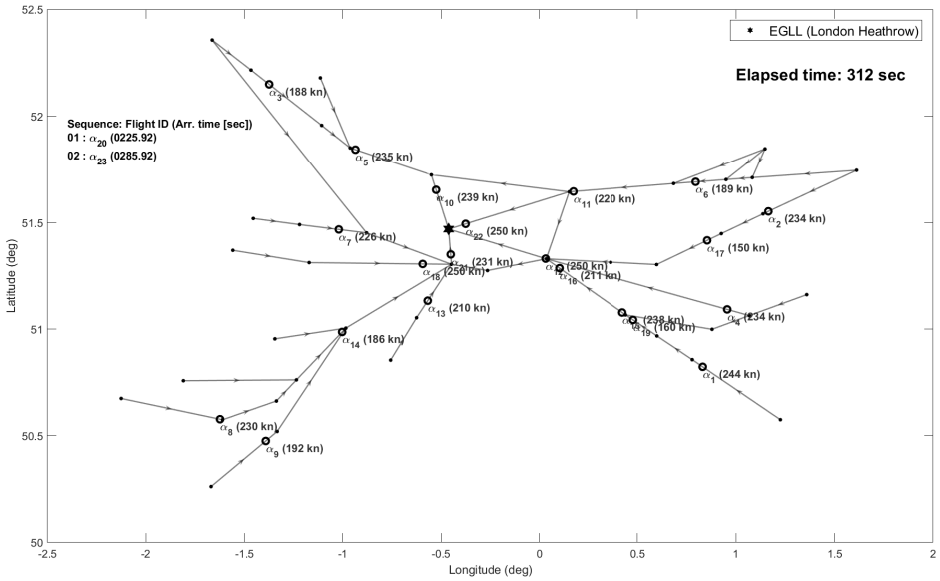


Figure 2.8: Case study results of 23 aircraft arriving at Heathrow airport in the LTMA: positions of aircraft at $t = 312$ sec

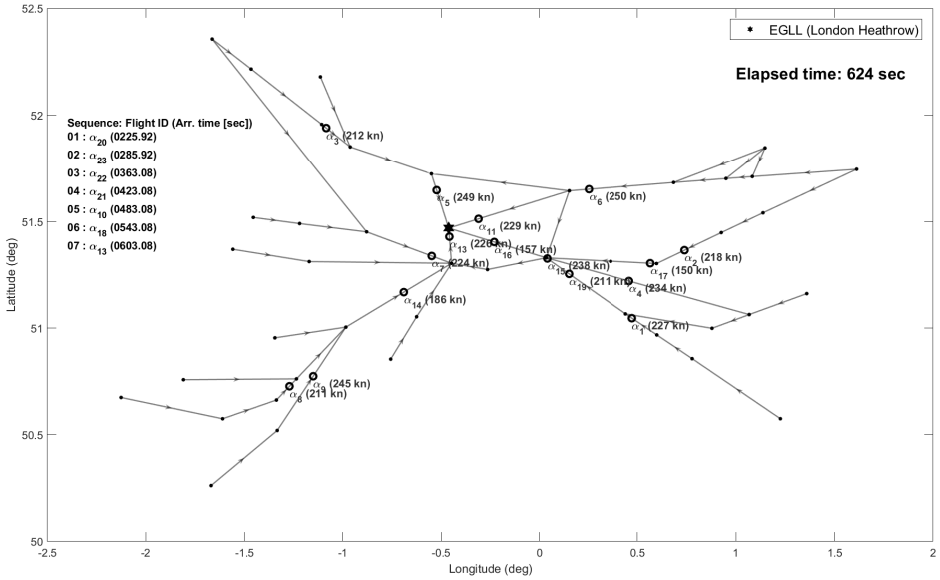


Figure 2.9: Case study results of 23 aircraft arriving at Heathrow airport in the LTMA: positions of aircraft at $t = 624$ sec

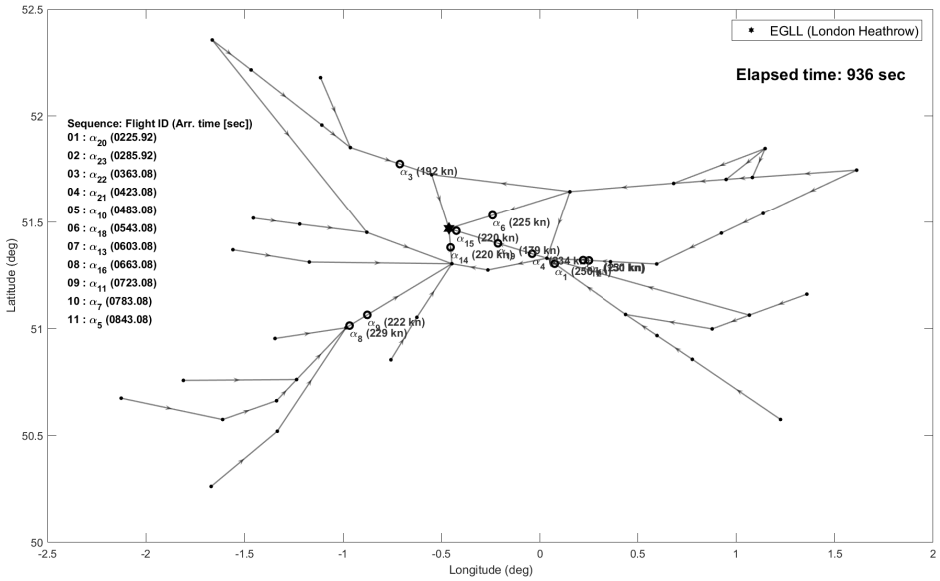


Figure 2.10: Case study results of 23 aircraft arriving at Heathrow airport in the LTMA: positions of aircraft at $t = 936$ sec

Table 2.3: Simulation results (arrival sequence, start waypoint (WP), merging waypoint (WP) and arrival time)

Flight ID	Arr. Seq.	Start WP	Merging WP	Arr. Time (sec)
α_{20}	1	BNN	BNN	226
α_{23}	2	DORKI	OCK	286
α_{22}	3	LAM	LAM	363
α_{21}	4	WOD	OCK	423
α_{10}	5	WCO	BNN	483
α_{18}	6	NIGIT	OCK	543
α_{13}	7	GWC	OCK	603
α_{16}	8	TIGER	BIG	663
α_{11}	9	BRASO	LAM	723
α_{7}	10	KENET	OCK	783
α_{5}	11	DTY	BNN	843
α_{12}	12	DET	BIG	903
α_{14}	13	BEGTO	OCK	963
α_{15}	14	LYD	BIG	1023
α_{6}	15	CLN	LAM	1083
α_{19}	16	ROTNO	BIG	1143
α_{3}	17	HON	BNN	1203
α_{4}	18	DVR	BIG	1263
α_{2}	19	LOGAN	LAM	1323
α_{1}	20	ALESO	BIG	1383
α_{17}	21	KOPUL	BIG	1443
α_{9}	22	DOMUT	OCK	1503
α_{8}	23	BILNI	OCK	1563

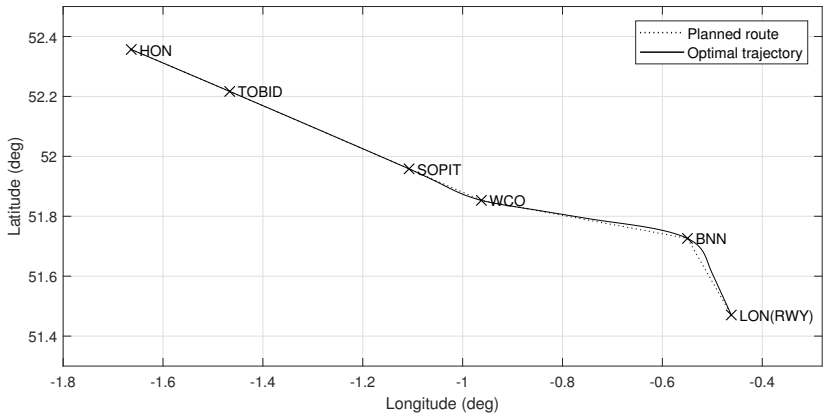


Figure 2.11: Optimised route based on α_3's flight route and schedule obtained from the proposed algorithm

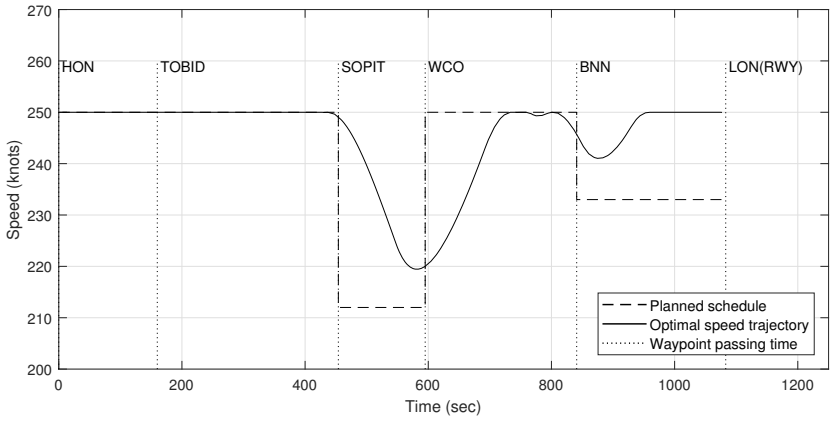


Figure 2.12: Optimised schedule based on α_3's flight route and schedule obtained from the proposed algorithm

Table 2.4: Simulation results (flight routes and speed profiles of 23 flights)

Flight ID	**Waypoint sequence (arrival time at each waypoint [sec])** **speed profile [km/h]**
α_1	ALESO(0) - ROTNO(345) - ETVAX(483) - TIGER(604) - BIG(936) - LON(1263) 250 - 250 - 250 - 238 - 224
α_2	LOGAN(0) - TRIPO(285) - SABER(372) - BRASO(553) - LAM(840) - LON(1203) 250 - 207 - 198 - 250 - 250
α_3	HON(0) - TOBID(160) - SOPIT(454) - WCO(595) - BNN(841) - LON(1083) 250 - 250 - 212 - 250 - 233
$\vdots$	$\vdots$
α_{21}	WOD(0) - OCK(266) - LON(423) 250 - 228
α_{22}	LAM(0) - LON(363) 250
α_{23}	DORKI(0) - OCK(110) - LON(286) 231 - 204

Table 2.5: RTFM arriving at London Heathrow airport (21:00 $\sim$ 21:20, Thursday 14 September 2017)

Flight ID	**Origin**	**Destination**	**Aircraft name (ICAO WTC)**	**Arr. Time (h:m)**
211116707	EGNT	EGLL	A319 (M)	21:00
211116002	EDDF	EGLL	A320 (M)	21:01
211114496	LEBL	EGLL	A320 (M)	21:02
211115949	EHAM	EGLL	A321 (M)	21:03
211114186	LPPT	EGLL	A320 (M)	21:05
211116372	EGPD	EGLL	A321 (M)	21:06
211116495	EGAC	EGLL	A319 (M)	21:09
211115728	LFML	EGLL	A320 (M)	21:10
211116249	EIDW	EGLL	A319 (M)	21:11
211114766	LIRF	EGLL	A321 (M)	21:12
211114520	EPWA	EGLL	B738 (M)	21:13
211114740	LEPA	EGLL	A321 (M)	21:15
211116925	EGPF	EGLL	A321 (M)	21:16
211116568	LFSB	EGLL	A319 (M)	21:17
211109711	OLBA	EGLL	A320 (M)	21:18

2.5 Conclusions

To deal with routing and scheduling problem, which are one of the most challenging problems in a TMA, a significant amount of research has been conducted. However, these approaches find a solution in a limited solution search space because these approaches solve the routing and scheduling problems separately or sequentially. At the later stages, therefore, optimality of the solution might be degraded. For multiple flights a new routing and scheduling algorithm has been presented that ensure minimum separation amongst inbound traffic. The proposed algorithm seeks the optimal flight route and its speed profile of each aircraft to maximise the runway throughput, given an airspace structure. The resultant outputs provide separation-compliant and speed-limitations-compliant flight routes and speed profiles. The main advantages of the proposed approach are three: solving the problem in a larger search space compared with the sequential approaches; efficient computational time; and providing separation-compliant segments and schedules. These advantages are relevant for the development of a new advance DST that helps ATCOs to manage traffic in a more efficient manner. The following additional studies with regards to the proposed algorithm are recommended to develop a full picture of a new advanced DST.

1. *Considering multiple airports*: For more realistic scenarios in TMAs with more than one airport such as the LTMA, an extended study should be conducted. In a TMA where there are multiple airports, we should consider crossing points as well as merging points.

2. *Considering various types of aircraft*: Since aircraft have different speed limitations and time-based separation requirements, it is necessary to consider all possible categories of aircraft types.

3. *Considering detailed flight from the holding stacks*: The considered assumptions should be reviewed to cope with a more detailed description of approaching procedures or separately considered in order to make the proposed algorithm more practical.

The three points will be tackled in the following section.

References

[1] SESAR Consortium, "European ATM Master Plan - Edition 2," *The Roadmap for Sustainable Air Traffic Management*, no. October, pp. 1–100, 2012.

[2] Federal Aviation Administration, "NextGen Update: 2014," Federal Aviation Administration, Tech. Rep. August, 2014.

[3] G. C. Carr, H. Erzberger, and F. Neuman, "Delay Exchanges in Arrival Sequencing and Scheduling," *Journal of Aircraft*, vol. 36, no. 5, pp. 785–791, sep 1999.

[4] A. Bayen, C. Tomlin, Yinyu Ye, and Jiawei Zhang, "An approximation algorithm for scheduling aircraft with holding time," in *2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601)*. IEEE, 2004, pp. 2760–2767 Vol.3.

[5] L. Bianco, P. Dell'Olmo, and S. Giordani, "Scheduling models for air traffic control in terminal areas," *Journal of Scheduling*, vol. 9, no. 3, pp. 223–253, jun 2006.

[6] H. Balakrishnan and B. Chandran, "Scheduling Aircraft Landings Under Constrained Position Shifting," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*. Reston, Virigina: American Institute of Aeronautics and Astronautics, aug 2006, p. 6320.

[7] Hanbong Lee and H. Balakrishnan, "A Study of Tradeoffs in Scheduling Terminal-Area Operations," *Proceedings of the IEEE*, vol. 96, no. 12, pp. 2081–2095, dec 2008.

[8] R. Chipalkatty, P. Twu, A. Rahmani, and M. Egerstedt, "Distributed scheduling for air traffic throughput maximization during the terminal phase of flight," *Proceedings of the IEEE Conference on Decision and Control*, pp. 1195–1200, 2010.

[9] A. V. Sadovsky, D. Davis, and D. R. Isaacson, "Separation-compliant, optimal routing and control of scheduled arrivals in a terminal airspace," *Transportation Research Part C: Emerging Technologies*, vol. 37, pp. 157–176, 2013.

[10] D. R. Isaacson, a. V. Sadovsky, and D. Davis, "Tactical Scheduling for Precision Air Traffic Operations: Past Research and Current Problems." *Journal of Aerospace Information Systems*, vol. 11, no. 4, pp. 234–257, 2014.

[11] A. V. Sadovsky, "Application of the Shortest-Path Problem to Routing Terminal Airspace Air Traffic," *Journal of Aerospace Information Systems*, vol. 11, no. 3, pp. 118–130, mar 2014.

[12] A. V. Sadovsky and M. M. Jastrzebski, "Strategic time-based metering that assures separation for integrated operations in a terminal airspace," *15th AIAA Aviation Technology, Integration, and Operations Conference*, no. June, pp. 1–14, 2015.

[13] R. Breil, D. Delahaye, L. Lapasset, and E. Feron, "Multi-agent systems for air traffic conflicts resolution by local speed regulation and departure delay," in *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, 2016, pp. 1–10.

[14] A. Rezaei, A. V. Sadovsky, and J. L. Speyer, "Existence and Determination of Separation-Compliant Speed Control in Terminal Airspace," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 6, pp. 1374–1391, jun 2016.

[15] S. Sidiropoulos, K. Han, A. Majumdar, and W. Y. Ochieng, "Robust identification of air traffic flow patterns in Metroplex terminal areas under demand uncertainty," *Transportation Research Part C: Emerging Technologies*, vol. 75, pp. 212–227, 2017.

[16] M. Samà, A. D'Ariano, P. D'Ariano, and D. Pacciarelli, "Scheduling models for optimal aircraft traffic control at busy airports: Tardiness, priorities, equity and violations considerations," *Omega (United Kingdom)*, vol. 67, pp. 81–98, 2017.

[17] A. Bayen, C. Tomlin, Yinyu Ye, and Jiawei Zhang, "MILP formulation and polynomial time algorithm for an aircraft scheduling problem," in *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*. IEEE, 2003, pp. 5003–5010.

[18] R. T. Wong, "Combinatorial Optimization: Algorithms and Complexity (Christos H. Papadimitriou and Kenneth Steiglitz)," *SIAM Review*, vol. 25, no. 3, pp. 424–425, jul 1983.

[19] V. V. Vazirani, *Approximation Algorithms*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003.

[20] H. Chida, C. Zuniga, and D. Delahaye, "Topology design for integrating and sequencing flows in terminal maneuvering area," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 230, no. 9, pp. 1705–1720, jul 2016.

[21] Eurocontrol, "Eurocontrol TBS Project: Aircraft Wake Vortex Modelling in support of the Time Based Separation Project, AO-06-11079EB," Eurocontrol, Tech. Rep., 2008.

[22] Eurocontrol, "Eurocontrol TBS Project: Results from the Decenber 2007 Time Based Separation Real-Time Simulation Exercises, EEC Report No. 411," Eurocontrol, Tech. Rep., 2009.

[23] C. Morris, J. Peters, and P. Choroba, "Validation of the time based separation concept at london heathrow airport," *Proceedings of the 10th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2013*, 2013.

[24] G. R. van Brummelen, *Heavenly Mathematics: The Forgotten Art of Spherical Trigonometry*, mar 2015.

[25] International Civil Aviation Organization, "Aircraft Operations. Volume I - Flight Procedures," International Civil Aviation Organization, Tech. Rep. October, 2006.

[26] Eurocontrol, "User Manual for the Base of Aircraft Data (Bada) Revision 3.12," Eurocontrol, Tech. Rep. August, 2014.

[27] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, dec 1959.

[28] M. Mesgarpour, C. N. Potts, and J. A. Bennell, "Models for aircraft landing optimization," *Proceedings of the 4th international conference on research in air transportation (ICRAT2010)*, pp. 1–4, 2010.

[29] Eurocontrol, "Demand Data Repository 2," 2017. [Online]. Available: https://www.eurocontrol.int/articles/ddr2-web-portal

[30] National Air Traffic Services, "Aeronautical Information Service," 2017. [Online]. Available: https://www.nats.aero/do-it-online/ais/

Chapter 3

Routing & Scheduling Algorithm for Multiple Heterogeneous Arrivals at Multiple Airports

In this chapter, we extend the algorithm proposed in Chapter 2. We expect the proposed algorithm to be more suitable for the Air Traffic Management (ATM) system by imposing the following considerations: heterogeneous aircraft; multiple airports; three-dimensional airspace; detailed flight route from the holding stacks to runways.

3.1 Introduction

A Terminal Manoeuvring Area (TMA) containing a complex set of route segments as shown in Figure 3.1 is a complex and dynamical system, which includes many operational constraints and uncertainties. The complexity comprises the operational procedures that constrain the lack of accuracy of the system, the heterogeneity in a variety of differently structured route networks near airports, and the human involvement in the control of the system. In today's air traffic operations in a controlled and structured airspace, it is essential to satisfy the prescribed minimum separation requirement for every pair of aircraft, where each aircraft tries to adhere as much as possible to a planned flight route and schedule unless Air Traffic Controllers (ATCOs) instruct any

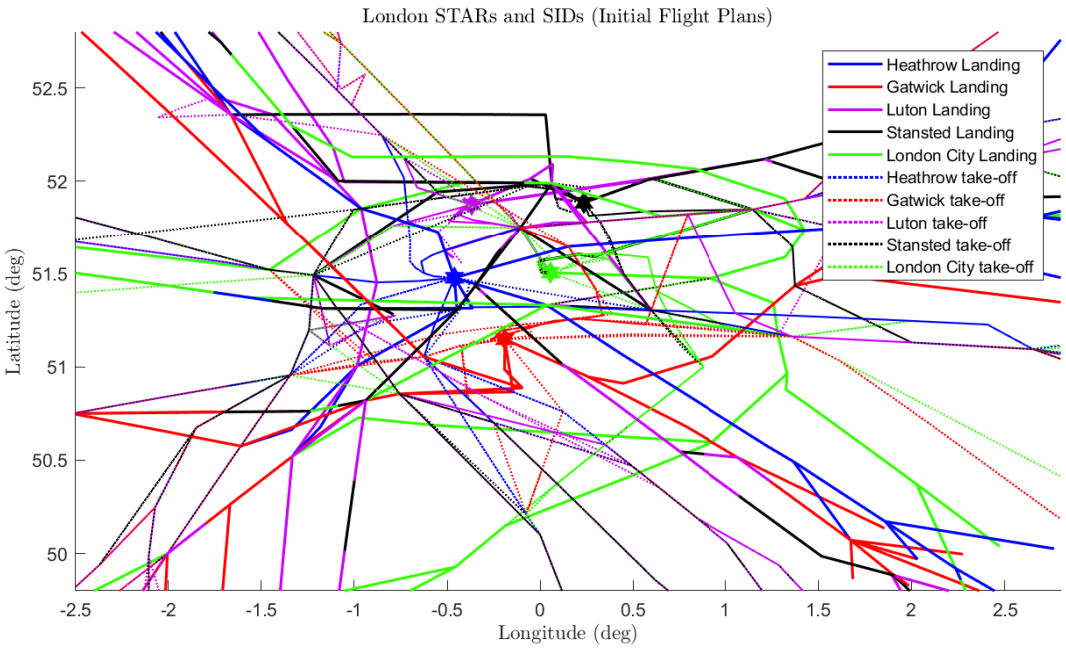


Figure 3.1: Regulated Tactical Flight Model (RTFM) based on London Standard Instrument Departure routes (SIDs) and STandard Arrival Routes (STARs) on 22 June 2017 obtained from Demand Data Repository (DDR2) [1]

change for safety and efficiency. ATCOs have responsibility for clearances that not only direct each aircraft's flight route (*path control*) but also its speed profile (*speed control*) with arrival sequencing to satisfy the separation requirement and to maintain the runway arrival capacity based on their flight plan. Managing multiple flights in a TMA is a computational problem whose complexity increases factorially with the number of flights in the area, but this becomes an issue especially during peak hours [2]. Human participation has the advantage that they can adaptively cope with unexpected situations. For example, ATCOs frequently change routes and schedules to improve efficiency and/or safety. However, this participation is based on some scheduling rules, the intuition, and experience of ATCOs without using formally defined performance indices. Such operations may lead to some inefficiencies on airspace and runway operations, and cannot be quantitatively evaluated for a runway operation analysis in terms of flight plans. The continuing growth in air traffic demand leads to increased number of sectors, as the system capacity is reaching its limit, and it is expected that the

Table 3.1: Operational goals pursued in the design of an automated decision support solution for the ATM system [3]

G1. operational safety
G2. schedule efficiency
G3. actual executability of the proposed solutions
G4. adaptability and robustness of the proposed solutions to unpredictable perturbations (e.g., wind or lapses in runway availability)
G5. correctness and sufficiently fast performance of the algorithms underlying the automation

demand will not be easily met without airspace principle changes and/or automation of the current system. To address these issues, the authors in [3] claim that decision support tools are needed, and list the conditions that should be met for the solution as listed in Table 3.1.

Definition 3 We define the Routing and Scheduling (R&S) problem as determining the following: (a) a separation-compliant flight route, (b) a separation-compliant speed profile including arrival times at each waypoint, (c) and an arrival sequence (if it is a multiple flights R&S problem).

To tackle a part of the R&S problem for multiple flights in a TMA, many studies have been conducted. In [4], the authors aimed to solve the routing problem based on the airspace graph so that the scheduling problem can be simplified. Then each schedule will be found based on the obtained flight routes. In [5], the authors proposed an algorithm for automating the process of speed control, which is divided into two stages; firstly, the algorithm finds an arrival sequence for the flights; secondly, the algorithm finds a collective speed profile of each flight for a given arrival sequence and

a predetermined flight route. In many studies, thus, the flight planning of a given set of flights in the airspace can also be considered as a two-part problem.

- Part 1: Routing problem (a discrete problem)
- Part 2: Scheduling problem (a continuous problem)

Although the computational complexity can be reduced by dividing the two problems, the solution search space can be reduced because the speed profile for Part 2 is calculated only based on the predetermined flight route in Part 1. In other words, it can not consider the other possible flight routes and sequence once those are determined, so it is likely that the results are inefficient. Furthermore, many approaches can only solve the problem with assumptions that are overly restrictive in real operational problems such as considering only one type of aircraft, a single airport, two-dimensional airspace network, detailed segments from Initial Approach Fixes (IAFs) to runway [3–11]. More recently, in [5], authors have conducted research to prove the existence and to determine separation-compliant speed profiles for multiple flights in a TMA. The authors leave the following as open problems: considering multiple airports; applying various types of aircraft; different separation requirements, etc. To sum up, the previous studies have considered the routing and scheduling problem separately or sequentially, and there are still practical issues remaining to implement to the ATM system. In this study, we focus on developing a multiple flights routing and scheduling algorithm capable of decision support for flight planning to comply with Table 3.1, and to complement the previous studies.

We first describe a graph-based problem formulation that tackles the R&S problem for a single flight, where we propose flight time weights for the graph. A solution of the proposed R&S problem provides a separation-compliant flight route and speed profile to the flight. By separation-compliant we mean that flights must fulfil the minimum separation time from a set of planned flights, as required by regulation. Secondly, we propose an algorithm for a multiple flights R&S problem based on the proposed single flight R&S formulation. The algorithm provides not only a solution of each flight R&S problem, but also an arrival sequence. The algorithm can address some of the open

problems presented in the previous studies such as multiple airports cases, detailed route segments from IAFs to runway, consideration of various types of aircraft with reasonable computational time. In addition to these features, the algorithm will find a solution for each flight R&S problem in a large solution search space.

We apply the proposed algorithm to an example of multiple arrivals at multiple airports in the London TMA to identify potential benefits of the algorithm in the flight planning phase. The simulation results show that the proposed algorithm can quickly provide separation-compliant flight routes, speed profiles, and an arrival sequence for the multiple flights.

The rest of this study is organised as follows. Section 3.2 describes the R&S problem formulation for a single flight. Section 3.3 describes the algorithm for solving the R&S problem for multiple flights. Numerical examples are given in Section 3.4 to present the optimality, to illustrate the process of the algorithm, and to demonstrate the advantages of the proposed algorithm. Section 3.5 concludes the study and discusses the further operational constraints to be implemented for the ATM system.

3.2 Routing and Scheduling Problem Formulation for a Single Flight

3.2.1 Routing Problem Modelling using a Weighted Digraph

For the operational purposes of the ATM system, airspace is often considered as a region with a set of waypoints. Some pairs of the waypoints are connected, and these are called (route) segments. In the TMA, STandard Arrival Routes (STARs) provide waypoints and routes with other information that airspace users should follow such as altitude constraints, track angle, etc. We utilise STARs to construct a nominal route network where aircraft fly over in the structured and controlled TMA.

Given an airspace network for the TMA, we firstly describe a routing problem where flights arrive at its destination from its initial waypoint. Note that only a single flight is considered in this section, although notations for the multiple flights are

used for consistency with later sections.

Definition 4 (Single Airport) Let $\mathcal{A} = \{\alpha_1, \alpha_2, \ldots, \alpha_i\}$ be a finite set of flights. The set is partitioned into two sets: a set of flights to be planned $\mathcal{A}^+$, and a set of planned flights $\mathcal{A}^-$ such that $\mathcal{A}^+ \cup \mathcal{A}^- = \mathcal{A}$, and $\mathcal{A}^+ \cap \mathcal{A}^- = \emptyset$.

(Multiple Airports) Let $\mathcal{B} = \{\beta_1, \beta_2, \ldots, \beta_b\}$ be a finite set of airports in a TMA. When there are more than two airports b, the flight set $\mathcal{A}^+$ is divided into the number of airports for each arrival airport: $\mathcal{A}_1^+, \mathcal{A}_2^+, \ldots, \mathcal{A}_b^+$ such that $\mathcal{A}_1^+ \cup \mathcal{A}_2^+ \cup \ldots \cup \mathcal{A}_b^+ = \mathcal{A}^+$, and $\mathcal{A}_b^+ \cap \mathcal{A}_{b'}^+ = \emptyset \, \forall \beta_b, \beta_{b'} \in \mathcal{B}, \forall b \neq b'$. Same to $\mathcal{A}_b^-$. Note that, α_i refers both the aircraft and the flight in this study.

Suppose that there exists a finite set $\mathcal{A}^+$ of arriving flights to be routed in the TMA. Each flight $\alpha_i \in \mathcal{A}^+$ has its initial waypoint and its destination in the TMA which are used for an input to the problem. The TMA is modelled as a directed graph (or digraph) $\mathcal{G} = (\mathcal{E}, \mathcal{V})$, called an *airspace graph* to highlight the aerospace context. In the airspace graph $\mathcal{G}$, vertex $v \in \mathcal{V}$ being a point in a Euclidean space of dimension two or three representing a waypoint candidate to be traversed. Edge $e \in \mathcal{E}$, corresponding a rectifiable curve, is a segment between some pair of waypoints in the airspace graph $\mathcal{G}$. In this model, a feasible flight route of the flight α_i in the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$ is defined as follows:

Definition 5 Given an airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$, flight $\alpha_i \in \mathcal{A}^+$ to be routed in $\mathcal{G}$, initial waypoints $v_1^i \in \mathcal{V}$ and destinations $v_{n_i}^i \in \mathcal{V}$, a flight route, denoted by $p(\alpha_i)$ in $\mathcal{G} = (\mathcal{E}, \mathcal{V})$, is defined as an ordered sequence of waypoints in $\mathcal{G} = (\mathcal{E}, \mathcal{V})$.

Thus, in the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$, there can be an abundance of flight route candidates denoted by $\mathcal{C}$ that satisfy the conditions as described in Definition 5. Through Definition 5, each flight route candidate $p(\alpha_i) \in \mathcal{C}$ can be given a corresponding set of waypoints $p(\alpha_i) : v_1^i, v_2^i, \ldots, v_{n_i}^i$, where n_i is the number of waypoints flight α_i traverses as shown in Figure 3.2.

Assumption 1 To reduce the complexity of the problem, the following flight related assumptions are made throughout this study: (a) each aircraft is able to exactly follow

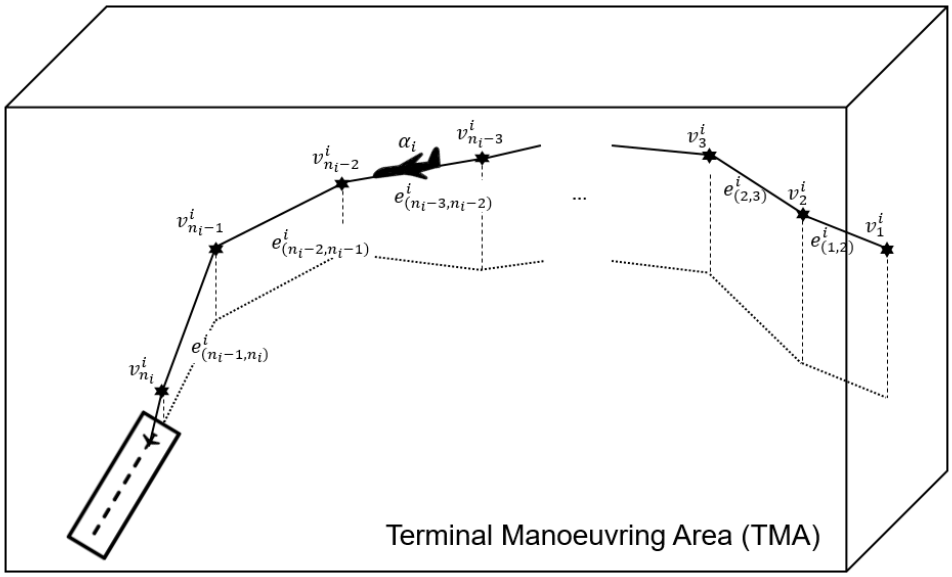


Figure 3.2: A series of linear segments (a flight route for flight α_i from its origin v^i_1 to destination $v^i_{n_i}$)

flight route candidates without deviation (i.e., aircraft are able to traverse cusps without deviation); (b) a speed profile of each flight (will be introduced later) is a set of constant values, where forms of the flight paths and speed profiles are the same as the flight routes and speed profiles in flight plans; and (c) the effect of uncertainties is neglected.

For each flight route $p(\alpha_i)$, a set of segments $\mathcal{E}(p(\alpha_i)) : e^i_{(1,2)}, e^i_{(2,3)}, \ldots, e^i_{(n_i-1,n_i)}$ exists, connecting waypoints through this flight route. The performance index for a set of segments $\mathcal{E}(p(\alpha_i))$ can be quantified as a weight set of positive numeric values $\mathcal{W}(p(\alpha_i)) : w^i_1, w^i_2, \ldots, w^i_{n_i-1}, \forall p(\alpha_i) \in \mathcal{C}$. Then, the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$ is transformed into a weighted digraph $\mathcal{G}_i = (\mathcal{E}, \mathcal{V}, \mathcal{W}_i), \forall \alpha_i \in \mathcal{A}^+$. We define that each flight has its own airspace graph as shown in Figure 3.3, because of different factors affecting the weight set for each airspace such as different initial positions, arrival airports, aircraft specifications.

Definition 6 Waypoint $v \in \mathcal{V}$ and segment $e \in \mathcal{E}$ are used $\forall \alpha_i \in \mathcal{A}$, whereas a set of weights $\mathcal{W}(p(\alpha_i))$ is a function of following (ownship), leading aircraft, and

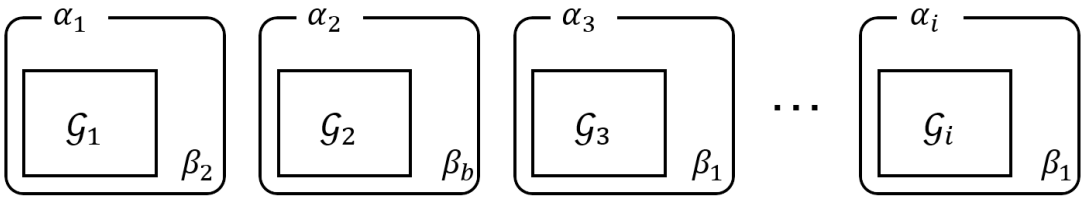


Figure 3.3: Airspace graphs for multiple aircraft arriving at multiple airports

permissible speed ranges of the aircraft. We define the unique airspace graph $\mathcal{G}_i = (\mathcal{E}, \mathcal{V}, \mathcal{W}_i)$, $\forall \alpha_i \in \mathcal{A}^+$.

For each airspace graph $\mathcal{G}_i = (\mathcal{E}, \mathcal{V}, \mathcal{W}_i)$, a performance index of each flight route can be estimated by summing all weighting values in $\mathcal{W}(p(\alpha_i))$ as follows:

$$\mathcal{T}(p(\alpha_i)) = \sum_{j=1}^{n_i - 1} w_j^i. \tag{3.1}$$

The flight routing problem that minimises the performance index can be defined as Definition 7.

Definition 7 Given the airspace graph $\mathcal{G}_i = (\mathcal{E}, \mathcal{V}, \mathcal{W}_i)$ and all corresponding flight route candidates $\mathcal{C}$, the flight routing problem is defined as finding a flight route such that

$$p^*(\alpha_i) = \underset{p(\alpha_i) \in \mathcal{C}}{\operatorname{argmin}} \mathcal{T}(p(\alpha_i)). \tag{3.2}$$

To solve this problem, well-known shortest path algorithms such as Dijkstra′s algorithm or A-star algorithm can be used [12].

Our main idea is that weights of the airspace graph could be expressed as flight times along routes that satisfy the minimum separation requirement from planned flights. Then, the obtained optimal flight route from the airspace graph for each flight implies the separation satisfied rectilinear segment and its weights sum is total flight time. In Section 3.2.2, we describe a horizontal separation concept built on time-based separation, which is necessary to propose a flight time weighting scheme for the R&S problem formulation.

3.2.2 Horizontal Separation Concept using Time-Based Separation

We take into account both vertical separation and horizontal separation. For the vertical separation requirement, we implement Reduced Vertical Separation Minimum (RVSM) that is 1000 ft used for IFR (Instrument Flight Rules) flight published in ICAO (International Civil Aviation Organization) documents. For the horizontal separation requirement, we propose a concept that complies with Time-Based Separation (TBS) under Assumption 2.

Assumption 2 In order to introduce the horizontal time-based separation concept the following assumptions are made throughout this study: (a) to satisfy safety requirement between each pair of aircraft, TBS is applied instead of the distance-based regime; (b) each flight's expected flight time data are shared; and (c) each flight is able to fly based on a given flight route and schedule.

TBS is designed to increase the runway arrival capacity and to improve the predictability of operation in headwind conditions [13, 14]. To introduce the horizontal separation concept, generic reference TBS is used, which is depending on the Wake Turbulence Category (WTC) of the leader and follower aircraft as shown in Table 3.2.

Table 3.2: Generic reference time-based separations [15]

Leader \ Follower	**A380 (*A*)**	**Heavy (*H*)**	**Medium (*M*)**	**Flight (*L*)**
A380 (*A*)	60s	145s	167s	189s
Heavy (*H*)	60s	98s	122s	145s
Medium (*M*)	60s	60s	60s	122s
Light (*L*)	60s	60s	60s	60s

Definition 8 Let $t_{SP}^{(i',i)}$, which is a function of the WTC of a leading aircraft $\alpha_{i'}$ and a following aircraft α_i, be a required separation time between $\alpha_{i'}$ and α_i, $\forall \alpha_i, \alpha_{i'} \in \mathcal{A}$, and $\forall i \neq i'$.

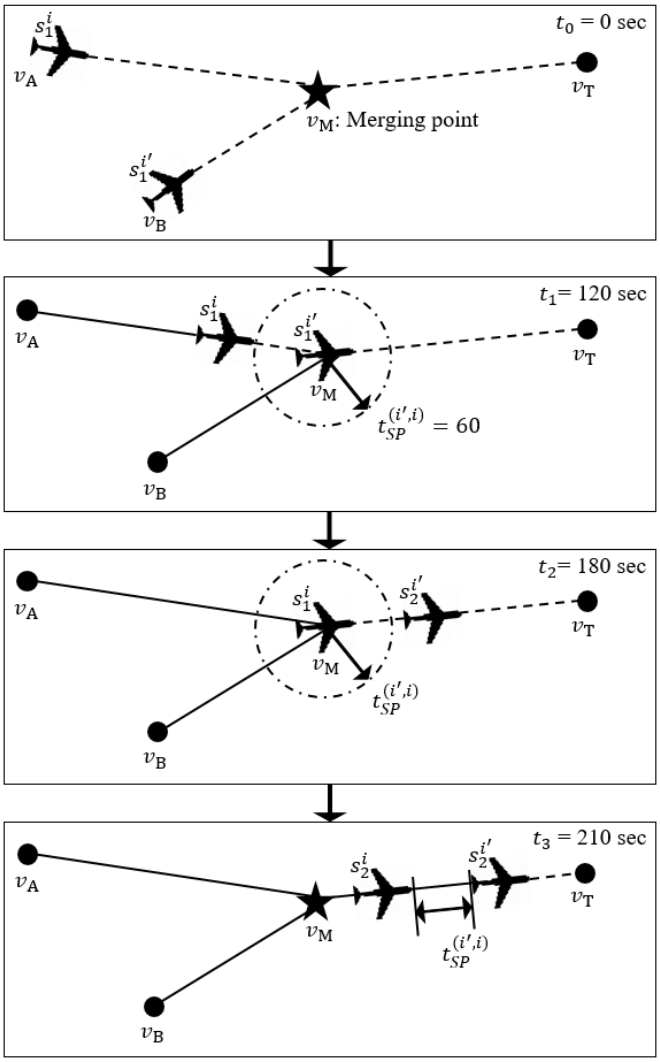


Figure 3.4: Time-based separation concept for merging points (heterogeneous flights, α_i, and $\alpha_{i'}$, fly from their origin v_A and v_B to the same destination v_T via the merging point v_M at the speed $\{s_1^i, s_2^i\}$ and $\{s_1^{i'}, s_2^{i'}\}$, respectively. The superscript and subscript of s are flight index and segment index, respectively)

Definition 9 Let $T(v), \forall v \in \mathcal{V}$ be ordered time data that contains the estimated time of arrival at waypoint v of aircraft $\alpha_i \in \mathcal{A}^-$ with its WTC. If none of the flights traverse over a waypoint v, then, $T(v) = \emptyset$.

For easy understanding, we describe the horizontal TBS concept with two examples. The concept is applicable to both crossing points and merging points as shown in

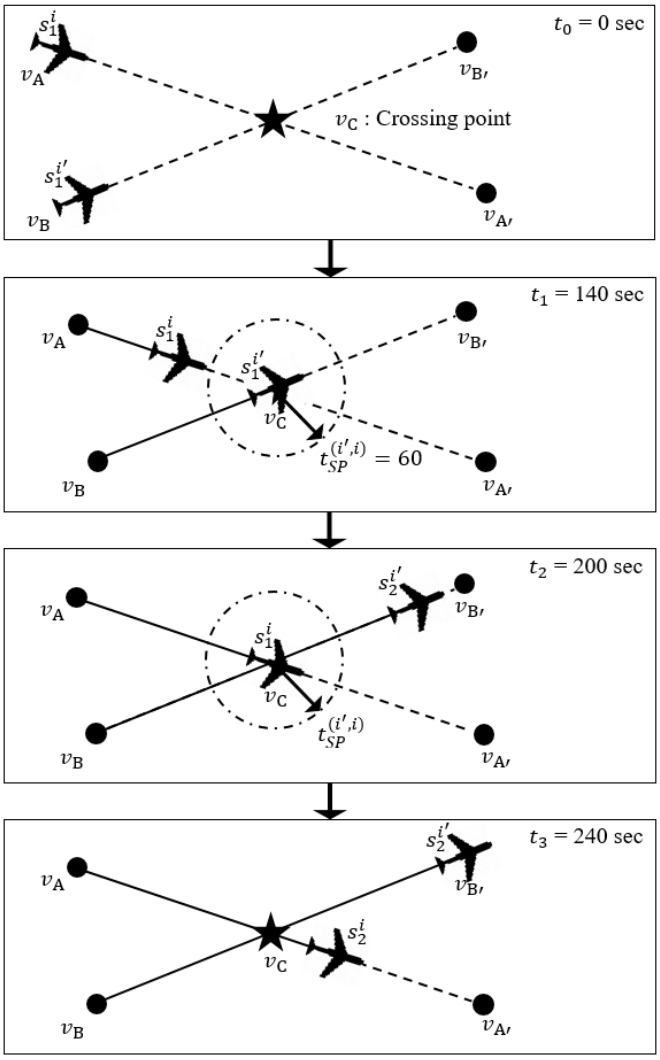


Figure 3.5: Time-based separation concept for crossing points (heterogeneous flights, α_i, and $\alpha_{i'}$, fly from their origin v_A and v_B to their destination $v_{A'}$ and $v_{B'}$ via the crossing point v_C at the speed $\{s_1^i, s_2^i\}$ and $\{s_1^{i'}, s_2^{i'}\}$, respectively. The superscript and subscript of s are flight index and segment index, respectively)

Figure 3.4 and Figure 3.5. Two medium WTC of aircraft α_i and $\alpha_{i'}$ traverse in simple airspace graphs. In Table 3.2, the horizontal separation requirement $t_{SP}^{(i',i)}$ is 60 seconds for both examples. In case of the merging point as shown in Figure 3.4, flight α_i and $\alpha_{i'}$ will fly from each initial waypoint v_{A} and v_{B} toward the same destination v_{T} via the same merging point v_{M}, respectively. Flight $\alpha_{i'}$ will pass through v_{M} at the speed

of $s_1^{i'}$ at time t_1 where the time t_1 stands for the Separation Assured Estimated Time of Arrival (SAETA). The SAETA, $t_1 = 60s$, is stored in the $T(v_{\mathrm{M}})$ with the WTC of aircraft $\alpha_{i'}$, which is M (medium). Then, no flight can pass through the merging point v_{M} for the required horizontal separation time $t_{SP}^{(i',i)} = 60$ seconds before and after the SAETA t_1. The SAETA t_2 with the WTC of α_i is also stored in the $T(v_{\mathrm{M}})$ in the same way for the following aircraft if there are following aircraft. Then, α_i and $\alpha_{i'}$ will traverse toward the destination v_{T} at the speeds of s_2^i, $s_2^{i'}$, respectively. In this example, we assume that there are only two flights. So that s_2^i and $s_2^{i'}$ are the maximum speed of each aircraft, and the horizontal separation requirement between the two aircraft is satisfied at all times. Finally, the time data at each waypoint is as follows:

$$T(v_{\mathrm{A}}) = \{0s(M)\}$$

$$T(v_{\mathrm{B}}) = \{0s(M)\}$$

$$T(v_{\mathrm{M}}) = \{60s(M), 180s(M)\}$$

$$T(v_{\mathrm{T}}) = \emptyset.$$

In the case of crossing points in Figure 3.5, the same process is applied as in the case of the merging point. The time data at each waypoint for the crossing point example are as follows:

$$T(v_{\mathrm{A}}) = \{0s(M)\}$$

$$T(v_{\mathrm{A'}}) = \emptyset$$

$$T(v_{\mathrm{B}}) = \{0s(M)\}$$

$$T(v_{\mathrm{B'}}) = \{240s(M)\}$$

$$T(v_{\mathrm{C}}) = \{140s(M), 200s(M)\}.$$

Such merging and crossing points can be easily found in STARs as shown in Figure 3.6.

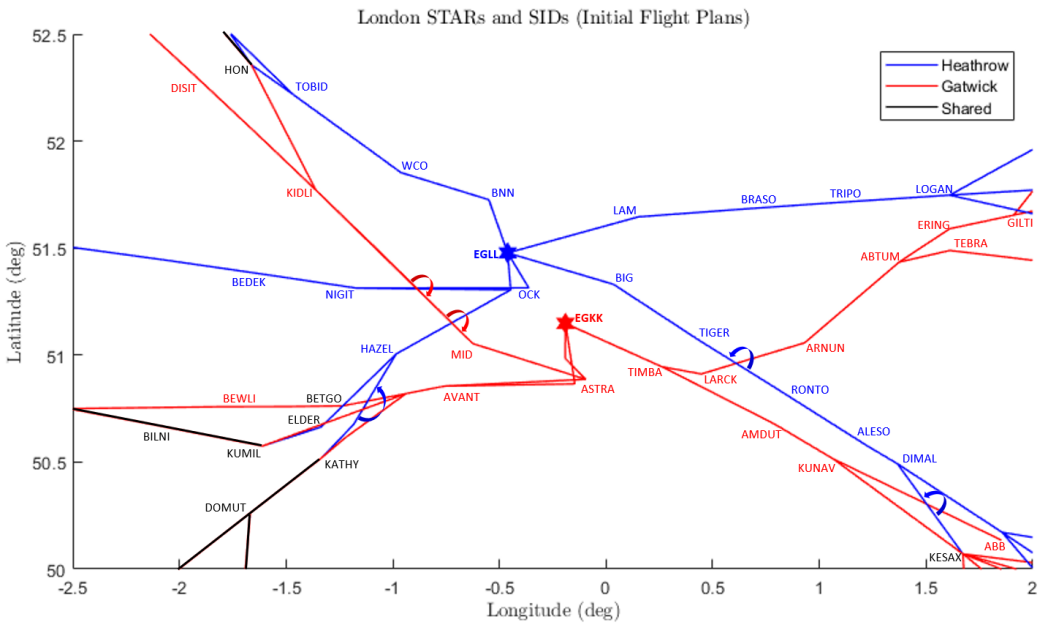


Figure 3.6: Airspace network to London Heathrow and Gatwick airport based on STARs and RTFM obtained from Demand Data Repository 2 [1]. The red lines indicate the segments to arrive at Gatwick airport, the blue lines indicate the segments to arrive at Heathrow airport, and the black lines indicate the shared segments to arrive at both airports. There are a few of unnamed crossing points denoted by the arrow arcs, which already satisfy vertical separation so airspace users do not need to consider that as the merging or crossing points. Only BETGO is the crossing point in this figure.

3.2.3 Flight Time Weighting Scheme

We propose a flight time weighting scheme $\mathcal{W}_{i,t}$ for the airspace graphs $\mathcal{G}_i$, $\forall \alpha_i \in \mathcal{A}^+$ as defined in Definition 6. First of all, we consider the time data $T(v)$ as a part of the airspace graph $\mathcal{G}_i = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}, T)$ subject to Assumption 2. The time data $T(v), \forall v$ becomes constraints for calculating $\mathcal{W}_{i,t}$. Secondly, to seek a separation-compliant speed profile, which is one of the purposes of the R&S problem defined in Definition 3, we introduce a set of constant speeds for each flight as follows:

$$s(\alpha_i) : s_1^i, s_2^i, \ldots, s_{n_i-1}^i$$

where s_i must be fitted into a feasible speed range ($[s_{\text{min}}^i \; s_{\text{max}}^i]$), $\forall \alpha_i \in \mathcal{A}$. The permissible speed range of each aircraft is depending on ICAO aircraft approach category [16].

Table 3.3 shows the Indicated AirSpeed (IAS) speed range for initial approach and final approach, in knots. Furthermore, we define a set of flight distances $d(\alpha_i)$ based on

Table 3.3: ICAO aircraft approach category

Aircraft category	**Range of speeds for initial approach [knots]**	**Range of speeds for final approach [knots]**	**Typical aircraft in this category**
A	90 - 150	70 - 110	small single engine
B	120 - 180	85 - 130	small multi engine
C	160 - 240	115 - 160	airline jet
D	185 - 250	130 - 185	large jet/military jet
E	185 - 250	155 - 230	special military

STARs allowing to compute speed profiles $s(\alpha_i)$, $\forall \alpha_i \in \mathcal{A}^+$

$$d(\alpha_i) : d_1^i, d_2^i, \ldots, d_{n_i-1}^i.$$

Let $u_i = 1/s_i$, $u_{\text{max}}^i = 1/s_{\text{min}}^i$, and $u_{\text{min}}^i = 1/s_{\text{max}}^i$, we formulate a linear programming problem and solve the problem to obtain weight $\mathcal{W}_{i,t}$, $\forall \alpha_i \in \mathcal{A}^+$ as follows:

$$\min \quad \sum_{j=1}^{n_i-1} u_j^i d_j^i \tag{3.3}$$

$$\text{s.t.} \quad d_1^i u_1^i + d_2^i u_2^i + \cdots + d_{n_i-1}^i u_{n_i-1}^i \geq \max(T(v_{n_i}^i)) + t_{SP}^{(i',i)} \tag{3.4}$$

$$d_1^i u_1^i + d_2^i u_2^i + \cdots + d_{n_i-2}^i u_{n_i-2}^i \geq \max(T(v_{n_i-1}^i)) + t_{SP}^{(i',i)} \tag{3.5}$$

$$\vdots$$

$$d_1^i u_1^i + d_2^i u_2^i \geq \max(T(v_3^i)) + t_{SP}^{(i',i)} \tag{3.6}$$

$$d_1^i u_1^i \geq \max(T(v_2^i)) + t_{SP}^{(i',i)} \tag{3.7}$$

$$u_{\text{min}}^i \leq u_1^i, u_2^i, \cdots, u_{n_i}^i \leq u_{\text{max}}^i \tag{3.8}$$

where s_{min}^i and s_{max}^i are aircraft minimum speed and maximum speed of aircraft α_i, respectively. Equation (3.3), which is the objective function, is to minimise sum of flight time of each segment subject to constraints of the separation requirement in Equations (3.4)-(3.7) at each waypoint v_j^i, $\forall j \in \{1, 2, \cdots, n_i - 1\}$ and the permissible speed range Equation (3.8), $\forall \alpha_i \in \mathcal{A}^+$. Time $t_{SP}^{(i',i)}$ is the required minimum separation between the following aircraft α_i and leading aircraft $\alpha_{i'}$ at each waypoint. We use only the SAETA and the WTC of the latest aircraft $\alpha_{i'} \in \mathcal{A}^-$ stored in $T(v_{n_i}^i)$.

The optimisation problem in Equations (3.3)-(3.8) can be solved by any well-known linear programming algorithm. In this study, we use the interior point method to solve the optimisation problem, and the result is used to assign flight time weights $\mathcal{W}_{i,t}$ for each flight route candidate of flight $\alpha_i \in \mathcal{A}^+$. Also, we calculate flight distance weights $\mathcal{W}_d$ in addition to the proposed flight time weights $\mathcal{W}_{i,t}$. The flight distance weights are used to determine priorities in case of more than two arrival flights at the same time at the same airport for multiple R&S problems. Positive numeric values of the physical segment distances are assigned to $\mathcal{W}_d$ that can be applied to all flights. Then, the airspace graph for each flight α_i is $\mathcal{G}_i = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}, \mathcal{W}_d, T)$. The single flight R&S problem that motivated this study can be finally formulated as follows:

Problem 2 Given an airspace graph $\mathcal{G}_i = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}, \mathcal{W}_d, T)$, flight α_i to be routed and scheduled in $\mathcal{G}_i$, and its initial waypoint $v_1^i \in \mathcal{V}$ and destination $v_{n_i}^i \in \mathcal{V}$, reachable from the origin, construct a flight route $p(\alpha_i)$ and a speed profile $s(\alpha_i)$, $\forall \alpha_i \in \mathcal{A}^+$ such that

- the horizontal separation requirement is satisfied from $\forall \alpha_i \in \mathcal{A}^-$,
- the speed profile $s(\alpha_i)$ must be within its feasible speed range $\forall \alpha_i \in \mathcal{A}$,
- and the airspace graphs are updated whenever flight $\alpha_i \in \mathcal{A}^+$ is routed and scheduled.

Therefore, the optimal flight route obtained by solving Problem 2 using any of well-known shortest path algorithms provides a solution for a single flight R&S problem. In this study, we utilise Dijkstra's algorithm to solve Problem 2 [17].

3.3 An Algorithm for Solving Routing and Scheduling Problem of Multiple Heterogeneous Aircraft Arriving at Multiple Airports

In the current ATM system, scheduling of the landing of flights can be divided into three stages: initial sequencing stage, modifying schedule stage, and freezing stage [18]. Generally, the initial sequencing stage is based on the FCFS algorithm, which is the landing order that would result, if each flight proceeded to the runway and landed without due consideration of other flights (i.e., no separation is considered). However, this causes many modifications and increased workload in the next two stages that follow.

In this section, we propose an algorithm to generate a solution to the multiple flights R&S problem defined in Definition 3 by iteratively solving the single flight R&S problem. The proposed algorithm solves the single flight R&S problems iteratively, and determines one flight at each iteration until all flights are planned. The FCFS algorithm is utilised for the determination of the order of the flights.

Remark 1 The FCFS algorithm accompanying Problem 2 in this study determines an arrival sequence as well as satisfies the separation requirement for the planned flights.

Remark 2 When there are multiple airports, there could exist crossing points and/or the shared segments as well as merging points as shown in Figure 3.5 and Figure 3.6. In this case, one of the factors that determines flight efficiency and the runway arrival throughput is a priority allocation of the airports. In Algorithm 2, we give priority to airport $\beta_b, \forall \beta_b \in \mathcal{B}$ that has the minimum interval between the last allocated flight $\alpha_{i'} \in \mathcal{A}_b^-$ and the flight to be allocated $\alpha_i \in \mathcal{A}_b^+$.

Initial waypoint, destination, ICAO WTC, and ICAO aircraft approach category for each flight $\alpha_i \in \mathcal{A}^+$ are the inputs of Algorithm 2. The output of the algorithm is a solution of the problems described in Definition 3, i.e., a separation-compliant flight route, a separation-compliant speed profile $\forall\alpha_i \in \mathcal{A}^+$, and an arrival sequence. Here, we describe an iteration process where a flight $\alpha_i \in \mathcal{A}^+$ is allocated at each iteration, until $\mathcal{A}^+$ is empty (Line 2). The algorithm generates $\mathcal{G}_i^k$ $\forall\alpha_i \in \mathcal{A}^+$. Then, each flight has its unique airspace graph that has flight time weights $\mathcal{W}_{i,t}^k$, which determine the flight route and its schedule (Line 3). Where the weights are depending on the separation requirements, aicraft types, segment distance, permissible speed ranges. The optimal flight route $p^*(\alpha_i)$ of each airspace graph $\mathcal{G}_i^k$ is obtained by solving the shortest path problem using Dijkstra's algortihm and each separation-compliant speed profile $s^*(\alpha_i)$ can be obtained by dividing the flight time weights $\mathcal{W}_{i,t}^k(p^*(\alpha_i))$ by flight distance $d(p^*(\alpha_i))$ (Line 4). As described in Remark 2, if there are more than two airports in the TMA, we select one of the airports. Then, a flight is allocated as k^{th} arrival flight amongst the candidates $\alpha_i \in \mathcal{A}_b^+$ using the FCFS algorithm (Line 5-7). If there is only one airport in the TMA, a flight is allocated as k^{th} arrival flight amongst the candidates $\alpha_i \in \mathcal{A}^+$ using the FCFS algorithm (Line 9). In case of that more than two candidates arriving at the same airport having the same SAETA, the flight distance weights $\mathcal{W}_d(p^*(\alpha_i))$ are utilised to allocate one flight as k^{th} arrival flight amongst the candidates having the same SAETA (Line 11-13). In any case, once α_i is allocated, Algorithm 2 removes α_i from $\mathcal{A}^+$, and stores α_i in $\mathcal{A}^-$ with its flight route and speed profile. The stored data is shared with flight $\alpha_i \in \mathcal{A}^+$ (Line 14-15 and Line 17-18). This process is continued until $\mathcal{A}^+$ is empty.

For easy understanding, we illustrate Figure 3.7 showing an example of the iteration process for multiple flights routing and scheduling where an airport is predetermined for each flight arrivals at multiple airport. (1^{st} iteration) α_2 and α_4 have the same arrival time to their airport, but α_2 is planned to arrive at β_b first because the time gap between α_2 and the preceding flight arriving at β_b is bigger than the other one, then α_2 is transferred from $\mathcal{A}_\beta^+$ to $\mathcal{A}_\beta^-$; (2^{nd} iteration) airspace graph $\mathcal{G}_i^2$ is updated, $\forall\alpha_i \in \mathcal{A}^+$, then α_A is planned to arrive at β_1 second, and transferred from $\mathcal{A}_1^+$ to $\mathcal{A}_1^-$; (3^{rd} iteration) airspace graph $\mathcal{G}_i^3$ is updated, $\forall\alpha_i \in \mathcal{A}^+$, then α_3 is planned to arrive at β_2 third and

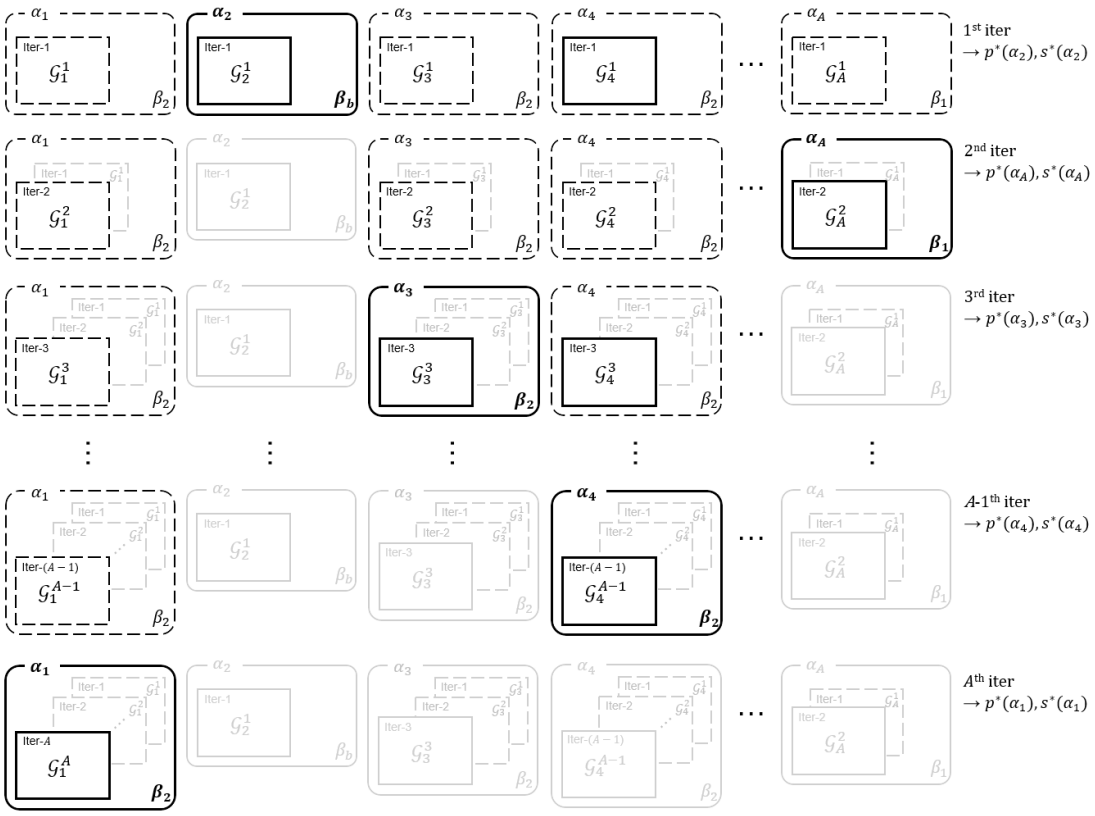


Figure 3.7: Iteratively generated airspace graphs for the multiple flights R&S problem

the flight is transferred from $\mathcal{A}_2^+$ to $\mathcal{A}_2^-$; ($(A-1)^{\text{th}}$ iteration) airspace graph $\mathcal{G}_i^{A-1}$ is updated $\forall \alpha_i \in \mathcal{A}^+$, then α_4 is planned to arrive at β_2 $A-1^{\text{th}}$, and the flight transferred from $\mathcal{A}_2^+$ to $\mathcal{A}_2^-$; (A^{th} iteration) airspace graph $\mathcal{G}_i^2$ is updated, $\forall \alpha_i \in \mathcal{A}^+$, then α_1 is planned to arrive at β_2 A^{th}, and transferred from $\mathcal{A}_2^+$ to $\mathcal{A}_2^-$. After the last iteration, we will have an empty set of $\mathcal{A}^+$, and $\mathcal{A}^- = \mathcal{A}$.

The proposed approach has the following advantages: (a) it finds a solution in a large solution space by simultaneously searching a flight route and schedule for each flight; (b) it is applicable to multiple airports; (c) it is applicable to various types of aircraft; (d) it is possible to apply the different minimum separation requirements according to the aircraft type. However, one disadvantage of the proposed algorithm is that arrival sequences must be solved in a centralised manner while the schedule for each flight is obtained in a decentralised manner. This may lead to a problem of increasing

the computational time. More discussion and future work are covered in Section 3.5.

Algorithm 2: R&S algorithm for multiple heterogeneous aircraft arriving at multiple airports

Input: $v_1^i, v_{n_i}^i \in \mathcal{V}$, ICAO WTC, and ICAO approach category $\forall \alpha_i \in \mathcal{A}^+$, STARs

Output: $p^*(\alpha_i), s^*(\alpha_i), \forall \alpha_i \in \mathcal{A}$ and the arrival sequence

$k = 1$

while $\mathcal{A}^+ \neq \emptyset$ **do**

  generate (update if $k > 1$) $\mathcal{G}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}^k, \mathcal{W}_d, T^k)$, $\forall \alpha_i \in \mathcal{A}^+$;

  find $p^*(\alpha_i), s^*(\alpha_i)$ of $\mathcal{G}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}^k, T^k)$, $\forall \alpha_i \in \mathcal{A}^+$ (Dijkstra's algorithm) ;

  if $b > 1$ **then**

    select $\mathcal{A}_b^+$ according to Remark 2;

    $\alpha_i^* \leftarrow \underset{\forall \alpha_i \in \mathcal{A}_b^+}{\operatorname{argmin}} p^*(\alpha_i)$ (FCFS algorithm);

  else

    $\alpha_i^* \leftarrow \underset{\forall \alpha_i \in \mathcal{A}^+}{\operatorname{argmin}} p^*(\alpha_i)$ (FCFS algorithm);

  end

  if *There are more than two* α_i^* *exist* **then**

    find $p^*(\alpha_i), s^*(\alpha_i)$ of $\mathcal{G}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_d, T^k)$ amongst them (Dijkstra's algorithm);

    $\alpha_i^\dagger \leftarrow \underset{\forall \alpha_i^*}{\operatorname{argmin}} p^*(\alpha_i)$ (FCFS algorithm);

    allocate $\alpha_i^\dagger$ as k^{th} arrival flight ;

    remove $\alpha_i^\dagger$ from $\mathcal{A}^+$, store $\alpha_i^\dagger$ in $\mathcal{A}^-$ with $p(\alpha_i^\dagger), s(\alpha_i^\dagger)$, and broadcast the flight plan to $\forall \alpha_i \in \mathcal{A}^+$;

  else

    allocate α_i^* as k^{th} arrival flight ;

    remove α_i^* from $\mathcal{A}^+$, store α_i^* to $\mathcal{A}^-$ with $p(\alpha_i^*), s(\alpha_i^*)$, and broadcast the flight plan to $\forall \alpha_i \in \mathcal{A}^+$;

  end

  $k = k + 1$;

end

3.4 Numerical Simulations

Three numerical examples are presented in this section. In the first example, we numerically show the optimality and scalability of the proposed algorithm by solving a small number of flights R&S problem in a simple airspace network. A second example is intended to illustrate the process of the proposed algorithm, and to compare the proposed algorithm with an existing algorithm. In a third example, a three-dimensional airspace network with 32 flights arrivals at two airports in the London TMA (LTMA) is used to illustrate the contributions of this study in a more realistic setting.

3.4.1 Optimality and Scalability

To show the optimality and computational efficiency of the proposed algorithm, we construct a simple two-dimensional airspace network including eight waypoints (L1∼L4, R1∼R4) and a merging point, and simulate seven cases while increasing the number of flights from two $\{\alpha_1, \alpha_2\}$ to eight $\{\alpha_1, \alpha_2, \cdots, \alpha_8\}$, as shown in Figure 3.8. Also, we illustrate the detail process of the algorithm using one of the seven cases in the first example. In these examples, we assume that every flight is a medium WTC (*M*) and ICAO aircraft category (C), which means that the minimum separation requirement between each pair of aircraft is 60 seconds and the speed range of each flight is from 160 knots to 240 knots.

Three arrival flights example to illustrate the algorithm

We consider a set of unplanned flights $\mathcal{A}^+ = \{\alpha_1, \alpha_2, \alpha_3\}$, and an empty set of planned flights $\mathcal{A}^- = \emptyset$ in the airspace network as shown in Figure 3.9.

In the first iteration, the algorithm generates an airspace graph for each flight considering the minimum separation requirement based on the time data of planned flights. As the algorithm generates airspace graphs based on the empty airspace, which means $T^1(v) = \emptyset$, $\forall v \in \mathcal{V}$, the shortest distance with maximum aircraft speed is the optimal flight route and schedule that maximises the runway capacity. A solution of each flight

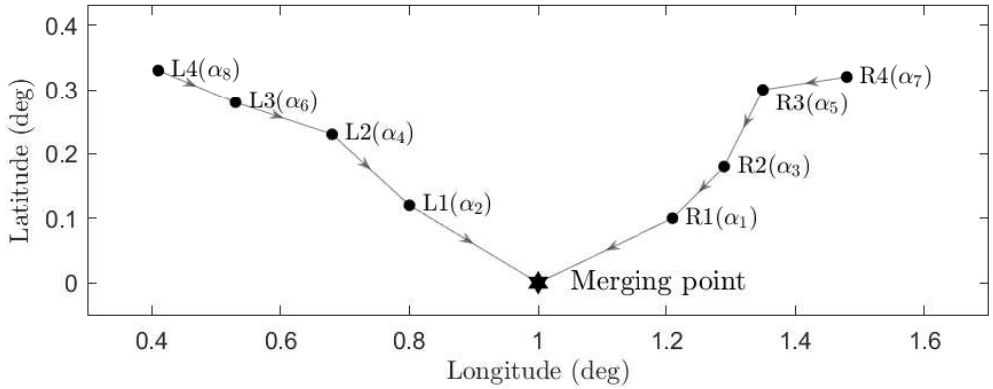


Figure 3.8: Configuration of eight flights in a simple airspace network

R&S problem in the first iteration is as shown in Table 3.4.

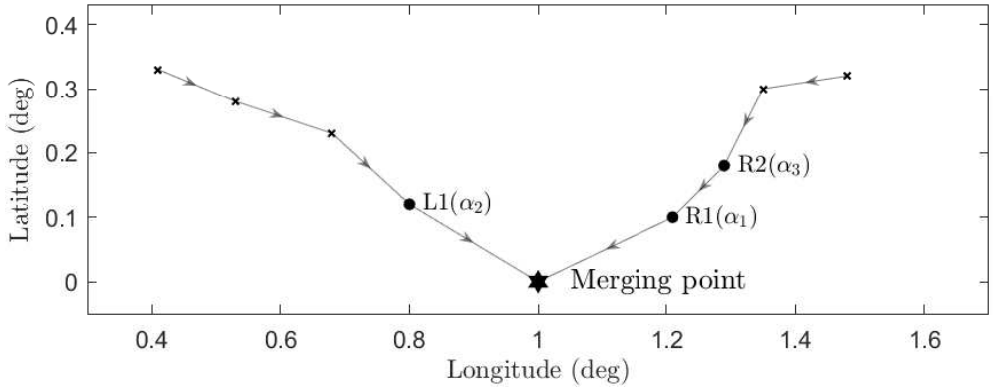


Figure 3.9: Configuration of three flights in a simple airspace network

Table 3.4: A separation-compliant flight plan for each flight in the first iteration

Flight ID	**Start waypoint**	**Waypoint sequence (arrival time [sec])**	**Speed profile [knot]**
α_1	R1	R1(0) - RW(209)	240
α_2	L1	L1(0) - RW(210)	240
α_3	R2	R2(0) - R1(102) - RW(311)	240 - 240

A priority will be given to the first arriving flight amongst the three candidates. Hence, flight α_1 is allocated to arrive the merging point first. Then, flight α_1 is removed from $\mathcal{A}^+ = \{\alpha_2, \alpha_3\}$ and stored in a set of planned flight $\mathcal{A}^- = \{\alpha_1\}$. In the second iteration, the time data T^2 is updated as follows

$$T^2(\text{Merging point}) = \{209s(M)\}$$
$$T^2(\text{R1}) = \{0s(M)\}$$
$$T^2(\text{L1}) = T^2(\text{R2}) = \emptyset$$

where the time data of α_1 is used to update T^2. Weights $\mathcal{W}^2_{i,t}$ of the airspace graph $\mathcal{G}^2_i$ is updated, $\forall \alpha_i \in \mathcal{A}^+$, as follows:

$$\mathcal{G}^2_2 = (\mathcal{E}, \mathcal{V}, \mathcal{W}^2_{2,t}, \mathcal{W}_d, T^2)$$
$$\mathcal{G}^2_3 = (\mathcal{E}, \mathcal{V}, \mathcal{W}^2_{3,t}, \mathcal{W}_d, T^2).$$

The optimal flight route and schedule for each flight in the second iteration is as shown in Table 3.5.

Table 3.5: A separation-compliant flight plan for each flight in the second iteration

Flight ID	**Start waypoint**	**Waypoint sequence (arrival time [sec])**	**Speed profile [knot]**
α_2	L1	L1(0) - RW(269)	187
α_3	R2	R2(0) - R1(119) - RW(329)	205 - 240

Using the FCFS algorithm, α_2 is secondly planned to arrive at the merging point. Then, α_2 is removed from the $\mathcal{A}^+ = \{\alpha_3\}$ and stored in the set of planned flights $\mathcal{A}^- = \{\alpha_1, \alpha_2\}$. In the third iteration, the time data T^3 is updated as follows:

$$T^3(\text{Merging point}) = \{209s(M), 269s(M)\}$$

$$T^3(\text{R1}) = T^3(\text{L1}) = \{0s(M)\}$$

$$T^3(\text{R2}) = \emptyset$$

where the flight plan of α_2 is used to update the time data T^3. Weights, $\mathcal{W}^3_{i,t}$ of the airspace graph $\mathcal{G}^3_i$ is updated, $\forall \alpha_i \in \mathcal{A}^+$, as follows:

$$\mathcal{G}^3_3 = (\mathcal{E}, \mathcal{V}, \mathcal{W}^3_{3,t}, \mathcal{W}_d, T^3).$$

The solution of flight α_3 R&S problem in the third iteration is shown in Table 3.6.

Table 3.6: A separation-compliant flight plan for each flight in the third iteration

Flight ID	Start waypoint	Waypoint sequence (arrival time [sec])	Speed profile [knot]
α_3	R2	R2(0) - R1(119) - RW(329)	204 - 240

As α_3 is the only remaining flights in $\mathcal{A}^-$, α_3 is selected as the third flight to arrive at the destination. The optimal flight route and schedule for each flight are given in Table 3.7.

Table 3.7: Final routes and speed profiles for the three flights

Flight ID	Start waypoint	Arrival sequence	Waypoint sequence (arrival time [sec])	Speed profile [knot]
α_1	R1	1	R1(0) - RW(209)	240
α_2	L1	2	L1(0) - RW(269)	187
α_3	R2	3	R2(0) - R1(119) - RW(329)	204 - 240

The outputs of the algorithm are the optimal flight route, schedule, separation-compliant speed profile and arrival sequence for each flight. If there are more flights to be planned following α_3, the time data T^4 would be updated as follows:

$$T^4(\text{Merging point}) = \{209s(M), 269s(M), 329s(M)\}$$

$$T^4(\text{R1}) = \{119s(M)\}$$

$$T^4(\text{L1}) = T^4(\text{R2}) = \{0s(M)\}$$

and the subsequent processes would be the same.

Optimality and scalability

We present numerical examples on the same airspace network illustrated in Figure 3.8 to compare the proposed algorithm with the exhaustive search algorithm. Through the comparison between results by increasing the number of flights, we show the optimality and scalability of the proposed algorithm numerically. Also, it is proven that the sub-algorithm, which is Dijkstra's algorithm, used in Algorithm 2 is a polynomial time algorithm [17]

The exhaustive search algorithm for finding a solution of the problem is as follows: (step 1) find all possible sequences of flights, which is factorial to the number of flights; (step 2) calculate the flight route and schedule for every flight for all the sequence cases; (step 3) choose one case that minimises the flight time of the last flight subject to satisfaction of the minimum separation requirement. Comparison results are given in Table 3.8.

The results show that the numerical optimality in terms of the last flight arrival time increases with the number of flights. An assumption of denying overtaking the leading flight on the same segment makes the solution of the proposed algorithm close to the solution of the exhaustive search algorithm. The results also show that the computational time increases polynomially with the number of flights while the exhaustive search's computational time is exponentially increasing.

3.4.2 Case Study: 32 Aircraft Arriving at Heathrow Airport and Gatwick Airport in the LTMA

In this example, the three-dimensional airspace network is a section of the London TMA (LTMA), with 32 flight ($\alpha_1 \sim \alpha_{16}$ to London Heathrow Airport (EGLL) and

Table 3.8: Optimality, computational time, and efficiency improvement of the proposed algorithm (PA: Proposed Algorithm, ESA: Exhaustive Search Algorithm, Effi.: Efficiency)

# of α_i	**Last α_i arrival time [sec]**		**Effi. [%]**	**Computational time [sec]**		**Effi. [%]**
	PA	**ESA**		**PA**	**ESA**	
2 (α_1, α_2)	269.48	269.48	100.00	0.0103	0.03	61.42
3 ($\alpha_1 \sim \alpha_3$)	329.48	315.09	95.43	0.0229	0.21	88.97
4 ($\alpha_1 \sim \alpha_4$)	389.48	375.09	96.16	0.0418	1.63	97.44
5 ($\alpha_1 \sim \alpha_5$)	449.48	435.09	96.69	0.0762	13.88	99.45
6 ($\alpha_1 \sim \alpha_6$)	509.48	499.07	97.91	0.1216	129.63	99.91
7 ($\alpha_1 \sim \alpha_7$)	569.48	559.07	98.14	0.1727	1422.17	99.99
8 ($\alpha_1 \sim \alpha_8$)	629.48	619.07	98.32	0.2419	16921.37	99.99

$\alpha_{17} \sim \alpha_{32}$ to London Gatwick Airport (EGKK)) flying in the airspace as shown in Figure 3.10, Figure 3.11 and Table 3.9. For the airspace network, we use Standard Terminal Arrival Route charts (STARs) from the United Kingdom Aeronautical Information Publication (AIP) [19]. We do not use all of the waypoints and segments listed in the STARs, but rather sort out nominal waypoints (WPs) and segments from EUROCONTROL Demand Data Repository 2 (DDR2) used for initial flight plannings [1]. Note that DDR2 consists of Filed Tactical Flight Model (FTFM), Regulated Tactical Flight Model (RTFM) and Current Tactical Flight Model (CTFM), and here we utilise the RTFM. The minimum separation requirement is depending on the WTC of the following and leading aircraft as given in Table 3.2, and the minimum vertical separation requirement is 1000 ft. For a more realistic simulation, we utilise 1.2 times increased permissible speed ranges than the ranges given in Table 3.3.

Assumption 3 (Holding points in the London TMA). In the London TMA, flights are routed to holding points (four holding points around Heathrow Airport, and two holding points around Gatwick Airport) where aircraft will hold until ATCOs are ready to

allocate them into an approach sequence to land. In this simulation, we assume that there are no holdings in order to demonstrate more general cases.

As the purpose of the proposed algorithm is to complement the current state-of-the-art of flight planning by providing flight routes and speed profiles that minimise the flight time of each flight, outputs of the algorithm are a flight route, separation-compliant speed profile for each flight, and arrival sequence. Table 3.10, and Table 3.11 shows the outputs for each flight route and its separation-compliant speed profile, and an arrival sequence, respectively. Figure 3.12 $\sim$ Figure 3.15 show the position of the flights every 500 seconds. The computation time of the algorithm for the example is about 4.46 seconds on a Windows 10 OS 3.4 GHz desktop computer with 16 GB RAM.

Table 3.9: Airspace graph and flight information

Airspace graph information		
Airport	London Heathrow (EGLL)	London Gatwick (EGKK)
# **of runways**	1 (landing only)	1 (landing only)
# **of WPs**	28 (incl. shared WPs)	27 (incl. shared WPs)
# **of segments**	27 (incl. shared segments)	27 (incl. shared segments)
Aircraft information		
# **of aircraft (H)**	3	3
# **of aircraft (M)**	11	12
# **of aircraft (L)**	2	1

Table 3.10: London TMA case simulation results (arrival sequence, start waypoint, and arrival time for each flight)

Flight ID (WTC)	**Possible Speed Range [knot]**	**Arrival Sequence (at Landing Airport)**	**Initial Position**	**Arrival time [sec]**
α_1 (M)	192 - 288	3 (EGLL)	LAM	373
α_2 (M)	192 - 288	7 (EGLL)	BRASO	676

α_3 (H)	222 - 300	14 (EGLL)	LOGAN	1319
α_4 (M)	192 - 288	13 (EGLL)	HON	1197
α_5 (M)	192 - 288	1 (EGLL)	H27R1	160
α_6 (M)	192 - 288	2 (EGLL)	H27R3	235
α_7 (M)	192 - 288	4 (EGLL)	OCK	433
α_8 (H)	222 - 300	5 (EGLL)	BNN	555
α_9 (M)	192 - 288	12 (EGLL)	KATHY	1137
α_{10} (L)	144 - 216	15 (EGLL)	KUMIL	1586
α_{11} (M)	192 - 288	6 (EGLL)	TIGER	616
α_{12} (M)	192 - 288	8 (EGLL)	HAZEL	737
α_{13} (M)	192 - 288	9 (EGLL)	NIGHT	797
α_{14} (L)	144 - 216	16 (EGLL)	DIMAL	1684
α_{15} (L)	144 - 216	10 (EGLL)	WCO	910
α_{16} (M)	192 - 288	11 (EGLL)	BEDEK	1032
α_{17} (M)	192 - 288	4 (EGKK)	LARCK	458
α_{18} (H)	222 - 300	5 (EGKK)	AMDUT	665
α_{19} (M)	192 - 288	14 (EGKK)	GILTI	1491
α_{20} (H)	222 - 300	10 (EGKK)	TEBRA	1186
α_{21} (H)	222 - 300	6 (EGKK)	MID	725
α_{22} (L)	144 - 216	16 (EGKK)	ABB	1805
α_{23} (M)	192 - 288	15 (EGKK)	DISIT	1652
α_{24} (M)	192 - 288	9 (EGKK)	ABTUM	1064
α_{25} (M)	192 - 288	13 (EGKK)	KIDLI	1431
α_{26} (M)	192 - 288	8 (EGKK)	KUNAV	885

α_{27} (M)	192 - 288	3 (EGKK)	TIMBA	367
α_{28} (M)	192 - 288	12 (EGKK)	DOMUT	1371
α_{29} (M)	192 - 288	2 (EGKK)	MAY	280
α_{30} (M)	192 - 288	11 (EGKK)	KESAX	1311
α_{31} (M)	192 - 288	7 (EGKK)	AVANT	825
α_{32} (M)	192 - 288	1 (EGKK)	G26L1	142

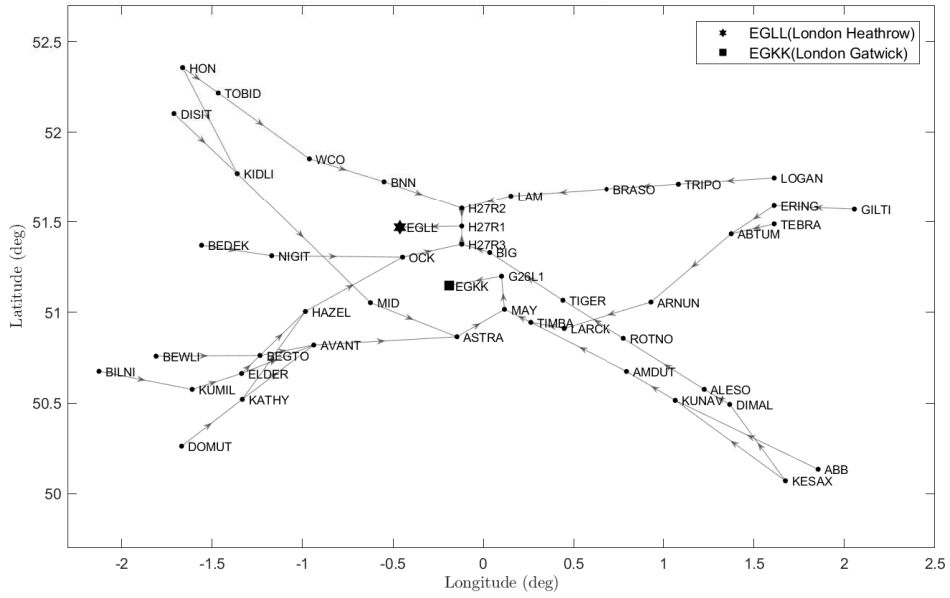


Figure 3.10: Airspace network with each waypoint name in the LTMA

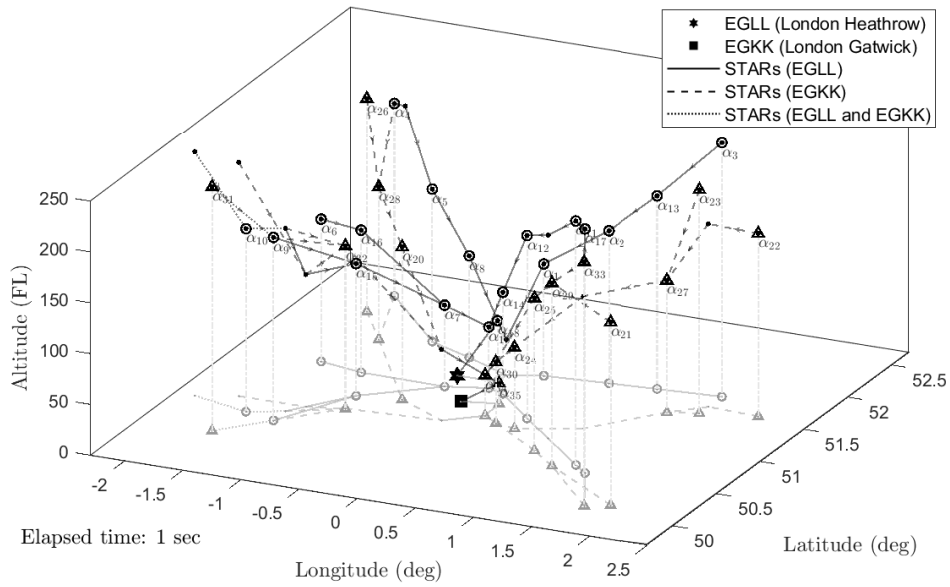


Figure 3.11: Initial position (three-dimensional) of 32 flights in the LTMA

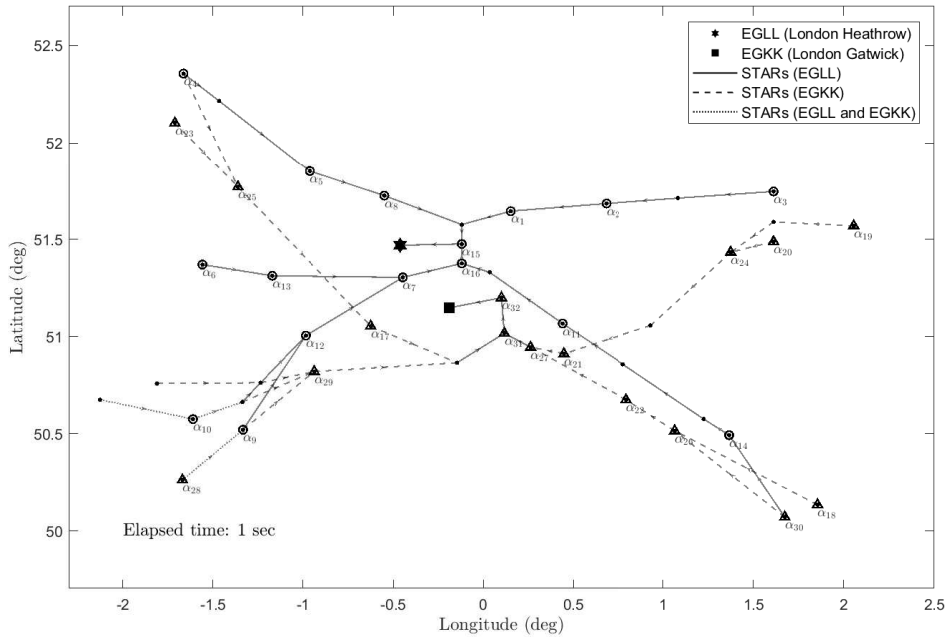


Figure 3.12: Case study results of 32 aircraft arriving at London Heathrow or Gatwick airport in the LTMA: positions of aircraft at $t = 1$ sec

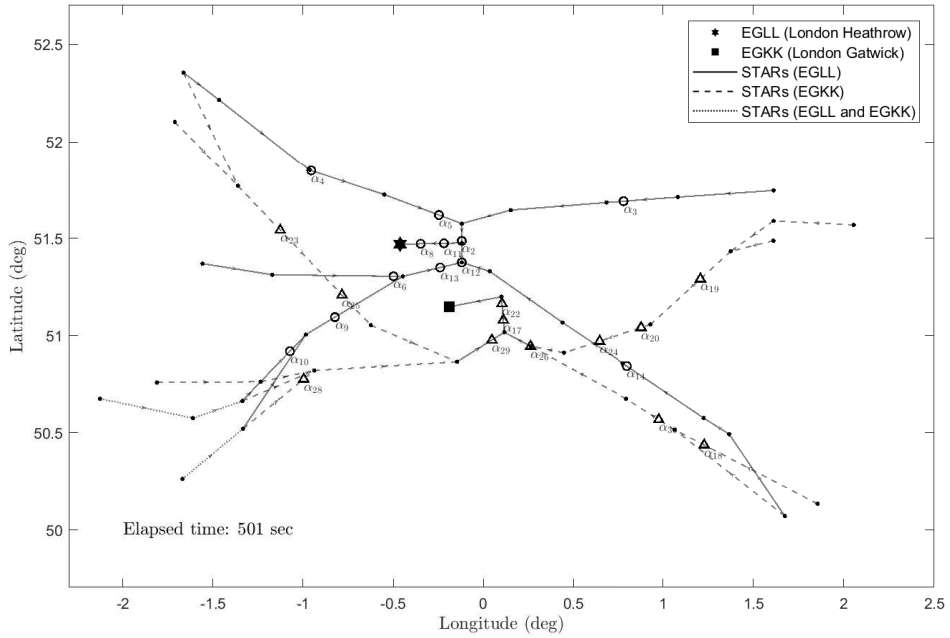


Figure 3.13: Case study results of 32 aircraft arriving at London Heathrow or Gatwick airport in the LTMA: positions of aircraft at $t = 501$ sec

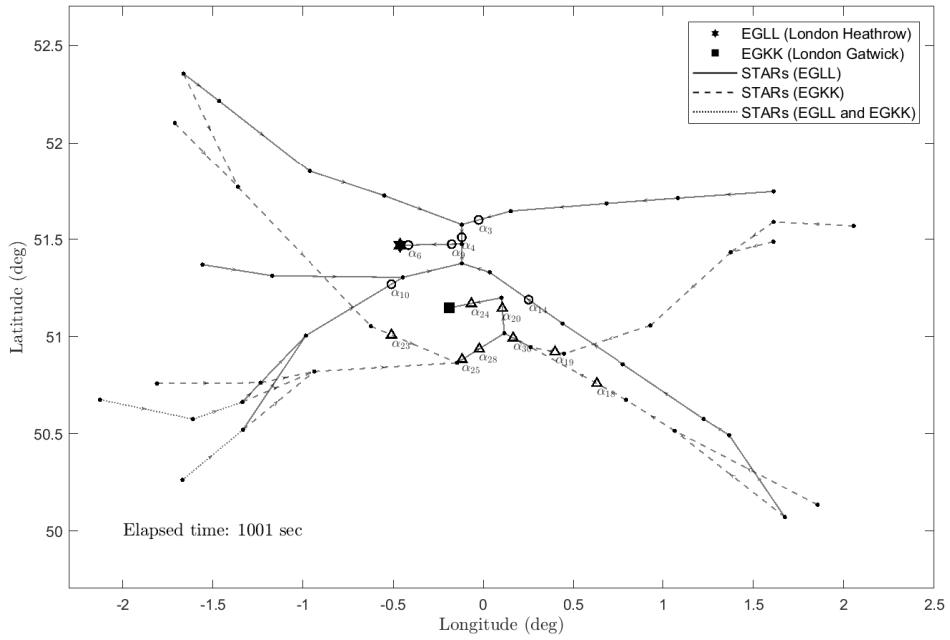


Figure 3.14: Case study results of 32 aircraft arriving at London Heathrow or Gatwick airport in the LTMA: positions of aircraft at $t = 1001$ sec

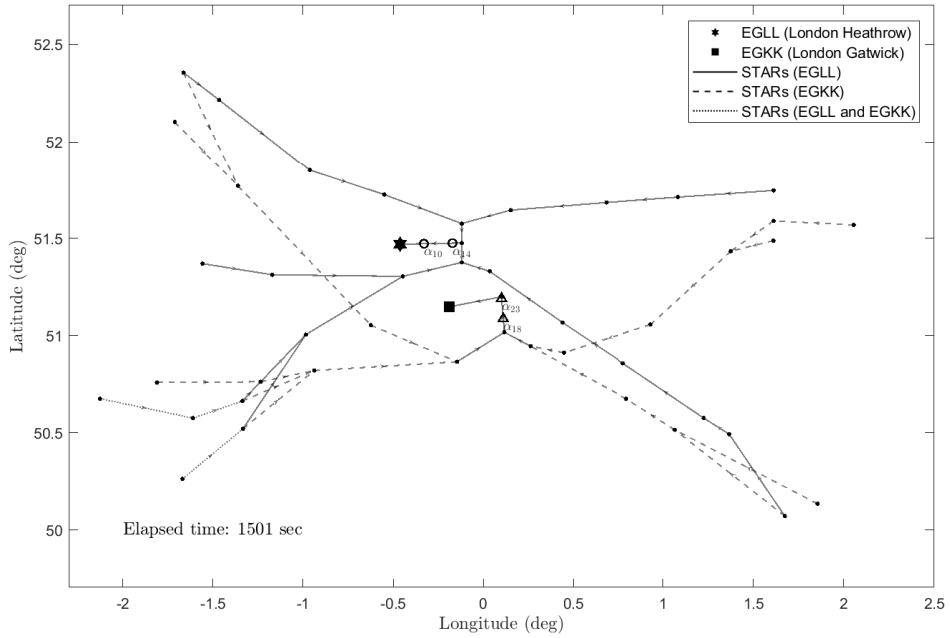


Figure 3.15: Case study results of 32 aircraft arriving at London Heathrow or Gatwick airport in the LTMA: positions of aircraft at $t = 1501$ sec

Table 3.11: London TMA case simulation results (flight route and speed profile for each flight)

Flight ID	**Flight route (arrival time [sec])** **Speed profile [knots]**
	To London Heathrow (EGLL)
α_1	LAM(0) - H27R2(138) - H27R1(213) - EGLL(373) 288 - 288 - 288
α_2	BRASO(0) - LAM(250) - H27R2(402) - H27R1(515) - EGLL(676) 288 - 259 - 191 - 288
	⋮
α_{15}	WCO(0) - BNN(286) - H27R2(592) - H27R1(696) - EGLL(910) 216 - 216 - 207 - 216
α_{16}	BEDEK(0) - NIGIT(187) - OCK(527) - H27R3(709) - H27R1(821) - EGLL(1032) 288 - 288 - 257 - 192 - 219
	To London Gatwick (EGKK)
α_{17}	LARCK(0) - TIMBA(91) - MAY(179) - G26L1(316) - EGKK(458) 288 - 288 - 288 - 288
α_{18}	AMDUT(0) - TIMBA(313) - MAY(396) - G26L1(528) - EGKK(665) 300 - 300 - 300 - 300
	⋮
α_{31}	AVANT(0) - ASTRA(377) - MAY(546) - G26L1(683) - EGKK(825) 288 - 288 - 288 - 288
α_{32}	G26L1(0) - EGKK(142) 288

3.5 Conclusions

This study discusses the value of improved adherence to the routes of the flight plans in the TMA. We propose the routing and scheduling algorithm, which provides separation-compliant flights and the arrival sequence for each flight, for multiple flights arrivals at multiple airports. We numerically compare the proposed algorithm with the exhaustive search algorithm to show its optimality and scalability. Also, we apply our algorithm to the case of the London TMA to show the benefits of the algorithm.

The results in this study show that the proposed algorithm provides a near-optimal solution to the problem, and its computational time increases polynomially with the number of flights. Although there are a few factors that are not yet covered in this study, it contains practical and detailed computed trajectories obtained by operational factors not considered in previous studies. Also, the algorithm finds the flight route and schedule in a larger solution search space than the existing approaches. It overcomes the limitations of the existing approaches, and the detailed trajectories could support decision making in the tactical phase by reducing the gap between the flight plans and actual flights as well as the flight planning phase.

Although this study overcomes some limitations of the previous studies, additional studies are required to utilise our algorithm as a practical decision-support tool in the ATM system.

1. *Aircraft speed*: In this study, we utilise IAS that contains errors such as instrument error. In order to apply the proposed algorithm to actual operations, it is required to implement ground speed.

2. *Considering both arrival and departure*: In general, arrival planning and departure planning are considered separately. However, the arrival flights must take into account departure flights for a more efficient operation, especially when there is only one runway. Further research can apply both arrival and departure planning by allocating predetermined departure sequence first based on the proposed algorithm.

3. *Applying to a special TMA*: Although our algorithm is suitable for the typical airports, we need further research to extend the algorithm to apply to particular cases. In the LTMA, for instance, there are a few holding stacks to manage the runway throughput. In this case, either introducing the holding stacks or separating the tactical stage into two segments (before the holding stacks, after the holding stacks).

4. *Including uncertainties*: Uncertainties produced by external sources such as wind and adverse weather have been discussed because it is directly related to safety. To consider the effect of such uncertainties, in general, conservative safety requirements have been implemented. Additional work can improve the efficiency of the ATM system by considering the wind effect where this effect can be approximated as quasi-static for the time-frame of the flight while in the TMA. Then, wind effect can be applied to the algorithm to update the computations of the R&S problem.

5. *Allowing new flights to enter the TMA for tactical planning*: The algorithm might consider new flights entering the TMA about 30-40 min before touchdown. To tackle the impact of this entering flights, a study may be conducted following the proposed algorithm. The number of flights entering the TMA is controlled by the air traffic flow management system, which works with characteristic times closely related to the computational time of the algorithm. The number of flights in the TMA should be planned with a maximum value. Therefore, the algorithm can be extended to be able to allow new entering flights only if there are an acceptable number of flights the algorithm can take.

References

[1] Eurocontrol, "Demand Data Repository 2," 2017. [Online]. Available: https://www.eurocontrol.int/articles/ddr2-web-portal

[2] A. V. Sadovsky, D. Davis, and D. R. Isaacson, "Efficient Computation of Separation-Compliant Speed Advisories for Air Traffic Arriving in Terminal Airspace," *Journal of Dynamic Systems, Measurement, and Control*, vol. 136, no. 4, p. 041027, may 2014.

[3] A. V. Sadovsky and M. M. Jastrzebski, "Strategic time-based metering that assures separation for integrated operations in a terminal airspace," *15th AIAA Aviation Technology, Integration, and Operations Conference*, no. June, pp. 1–14, 2015.

[4] A. V. Sadovsky, "Application of the Shortest-Path Problem to Routing Terminal Airspace Air Traffic," *Journal of Aerospace Information Systems*, vol. 11, no. 3, pp. 118–130, mar 2014.

[5] A. Rezaei, A. V. Sadovsky, and J. L. Speyer, "Existence and Determination of Separation-Compliant Speed Control in Terminal Airspace," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 6, pp. 1374–1391, jun 2016.

[6] A. V. Sadovsky, D. Davis, and D. R. Isaacson, "Separation-compliant, optimal routing and control of scheduled arrivals in a terminal airspace," *Transportation Research Part C: Emerging Technologies*, vol. 37, pp. 157–176, 2013.

[7] D. R. Isaacson, a. V. Sadovsky, and D. Davis, "Tactical Scheduling for Precision Air Traffic Operations: Past Research and Current Problems." *Journal of Aerospace Information Systems*, vol. 11, no. 4, pp. 234–257, 2014.

[8] R. Breil, D. Delahaye, L. Lapasset, and E. Feron, "Multi-agent systems for air traffic conflicts resolution by local speed regulation and departure delay," in *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, 2016, pp. 1–10.

[9] S. Sidiropoulos, K. Han, A. Majumdar, and W. Y. Ochieng, "Robust identification of air traffic flow patterns in Metroplex terminal areas under demand uncertainty," *Transportation Research Part C: Emerging Technologies*, vol. 75, pp. 212–227, 2017.

[10] S. Bae, H.-S. Shin, C.-H. Lee, and A. Tsourdos, "A New Multiple Flights Routing and Scheduling Algorithm in Terminal Manoeuvring Area," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*. IEEE, sep 2018, pp. 1–9.

[11] S. Bae, H.-S. Shin, and A. Tsourdos, "A New Graph-Based Flight Planning Algorithm for Unmanned Aircraft System Traffic Management," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*. IEEE, sep 2018, pp. 1–9.

[12] R. T. Wong, "Combinatorial Optimization: Algorithms and Complexity (Christos H. Papadimitriou and Kenneth Steiglitz)," *SIAM Review*, vol. 25, no. 3, pp. 424–425, jul 1983.

[13] Eurocontrol, "Eurocontrol TBS Project: Aircraft Wake Vortex Modelling in support of the Time Based Separation Project, AO-06-11079EB," Eurocontrol, Tech. Rep., 2008.

[14] Eurocontrol, "Eurocontrol TBS Project: Results from the Decenber 2007 Time Based Separation Real-Time Simulation Exercises, EEC Report No. 411," Eurocontrol, Tech. Rep., 2009.

[15] C. Morris, J. Peters, and P. Choroba, "Validation of the time based separation concept at london heathrow airport," *Proceedings of the 10th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2013*, 2013.

[16] International Civil Aviation Organization, "Aircraft Operations. Volume I - Flight Procedures," International Civil Aviation Organization, Tech. Rep. October, 2006.

[17] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, dec 1959.

[18] M. Mesgarpour, C. N. Potts, and J. A. Bennell, "Models for aircraft landing optimization," *Proceedings of the 4th international conference on research in air transportation (ICRAT2010)*, pp. 1–4, 2010.

[19] National Air Traffic Services, "Aeronautical Information Service," 2017. [Online]. Available: https://www.nats.aero/do-it-online/ais/

Chapter 4

Flight Planning Algorithm for Drone Delivery Services

In this chapter, we apply the algorithm proposed in Chapter 2 into Unmanned Aircraft System (UAS) Traffic Management (UTM) applications. For UTM applications, we modify the algorithm to suit this complex urban airspace, since more complex route network-based urban airspace operations are expected.

4.1 Introduction

In the near future, small Unmanned Aircraft Systems (sUASs) are expected to be integrated with other airspace users for different purposes, e.g. shipping with sUASs might be more efficient than shipping with conventional vehicles [1], sUASs might transport medical supplies and necessaries over hazardous terrain during a state of emergency [2]. There is a need for the safe operation and the efficient integration with current airspace users. Recently, the Federal Aviation Administration (FAA) published the regulation to allow the operation of sUASs in the National Airspace System [3]. To avoid interference with manned aircraft operations, the operations in the regulation limit altitude in 400 feet above ground level or within 400 feet of a structure, maximum weight to 55 lbs. (25 kilograms), the maximum ground speed of 100 mph (87 knots), etc. More details of the operational limitations are listed in [3]. A variety of

international and national organisations have initiated projects on Unmanned Aircraft System (UAS) Traffic Management (UTM) system to establish policies, requirements, frameworks, and infrastructure for the safe operation and the efficient integration [4–7]. The projects commonly emphasise that the UTM system requires services such as geo-fencing, airspace design, routing (route planning), scheduling, separation management (spacing and sequencing), contingency management as in the case of current Air Traffic Management (ATM) system. The UTM operations and services might be affected by the current ATM system which is the most relevant and reliable system.

Amongst the services, demands of routing, scheduling, and separation management have been raised because not only because of its primary aim but also these can be utilised as one of the factors determining airspace capacity and throughput of the service points. Many studies in the literature have focused on the sUAS path planning problem, and solved the problem with free flight or free routing approaches [8–13]. In high-density urban airspaces, the free flight-based operation may have difficulty in the safety, flight priority, and contingency operations. To consider potential interactions with the other users, and infrastructure, one possible approach is route network-based operations. In [7], authors focus on the route networks and its capacity and throughput. The capability and throughput are analysed by using a flight planning algorithm that only considers constant aircraft speed, and is time-independent, i.e., the authors solved the routing problem with an assumption that once a sUAS occupies a segment, then any other sUASs cannot traverse the occupied segment. However, the approach causes unrealistic urban airspace capacity analysis results because of conservative assumptions such as constant speed, non-temporal separation. It might degrade the optimality of solutions, although these have a computational advantage. In the ATM system, the routing, scheduling, and separation management are tackled separately or sequentially. In tactical phases, especially, Air Traffic Controllers (ATCOs) make decisions manually based on their experience, intuition and some rules without using formally defined performance indices [14].

In this study, we extend an algorithm, which we proposed in [15, 16] for the multiple aircraft routing and scheduling for commercial aircraft, to solve the route network-based flight planning problems (routing and scheduling) for multiple sUASs in high-

density urban airspaces also called Very Low Level (VLL) airspace. The algorithm generates each sUAS' flight plan iteratively. At each iteration, each sUAS finds its flight plan in a decentralised way by solving a graph-based problem we formulate (inner loop), and one of the sUASs is allocated using centralised algorithms (outer loop). The outer loop continues until all sUASs are planned. For the inner loop, we utilise a flight time weighting scheme to minimise each sUAS' flight time that satisfies the time-based separation requirement. By assigning the weights in the graph, we can consider a few practical factors such as feasible speed ranges of the sUASs, different separation requirements, etc. Then, each sUAS has its unique VLL airspace graph, and its optimal flight route obtained by using any well-known shortest path algorithm implies each sUAS' flight plan. We focus not only proposing the flight planning algorithm for multiple flights over the route networks, but also a comparison between two different outer loop algorithms. The algorithm is demonstrated on the last mile delivery (1-to-$\mathcal{M}$) and the first mile delivery ($\mathcal{N}$-to-1) cases. Results of the two case studies demonstrate the multiple sUASs' flight planning and provide insights into establishing detailed requirements for the sUAS operations in the high-density VLL airspace. Throughout this preliminary study, we expect that more efficient route network-based flight planning, and an analysis for the high-density VLL urban airspace.

The rest of this study is organised as follows. Section 4.2 formulates the problem for a single sUAS flight planning. Section 4.3 describes the proposed algorithm for multiple sUASs' flight planning problem. As a preliminary case study, over-road route networks for "last-mile delivery" and "first-mile delivery" are given and solved in Section 4.4, followed by concluding remarks in Section 4.5.

4.2 Problem Statement for a Single sUAS Flight Planning

Our focus is on flight planning a finite set $\mathcal{A}^+ = \{\alpha_1, \alpha_2, \ldots, \alpha_i\}$ of sUASs in the high-density VLL airspace where each sUAS $\alpha_i \in \mathcal{A}^+$ traverses from its origin to its destination. The flight planning, in this context, means finding routes, its speed profile (schedule), and departure or an arrival sequence while satisfying the separation

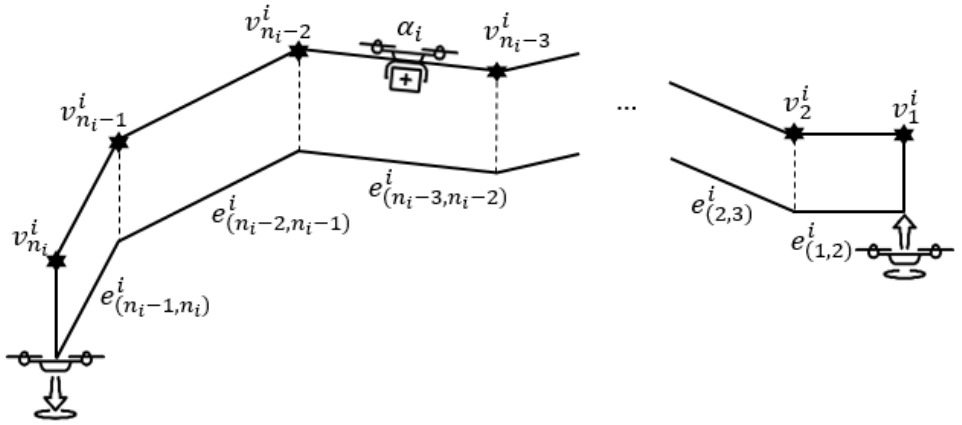


Figure 4.1: A series of linear segments (sUAS α_i's route from its origin v^i_1 to destination $v^i_{n_i}$)

requirement from a finite set $\mathcal{A}^-$ of planned sUASs.

Main assumptions we make for this study are as follows: each flight route is composed of a series of linear route segments as shown in Figure 4.1; each sUAS traverses along each segment at a constant speed; each UASs can accurately follow the segment without deviation; a time-based separation concept is utilised; uncertainties produced by external sources are neglected such as adverse weather; flight plans are shared between sUASs; the hovering manoeuvre is not considered.

4.2.1 Separation Concept in Route Network-Based Unban Airspace for Homogeneous sUASs

Conflict detection is activated when separation of two sUASs is less than a minimum separation criterion. In this study, we apply the time-based separation concept instead of the widely used distance-based separation concept to stabilise the spacing between the sUASs. By satisfying the separation assurance flight time for each waypoint rather than adjusting the distance between sUASs, we can obtain results that satisfy the minimum separation at segments as well as at waypoints at all times. The concept allows time-based separation to be satisfied at both merging points and crossing points as described in Section 4.2.2.

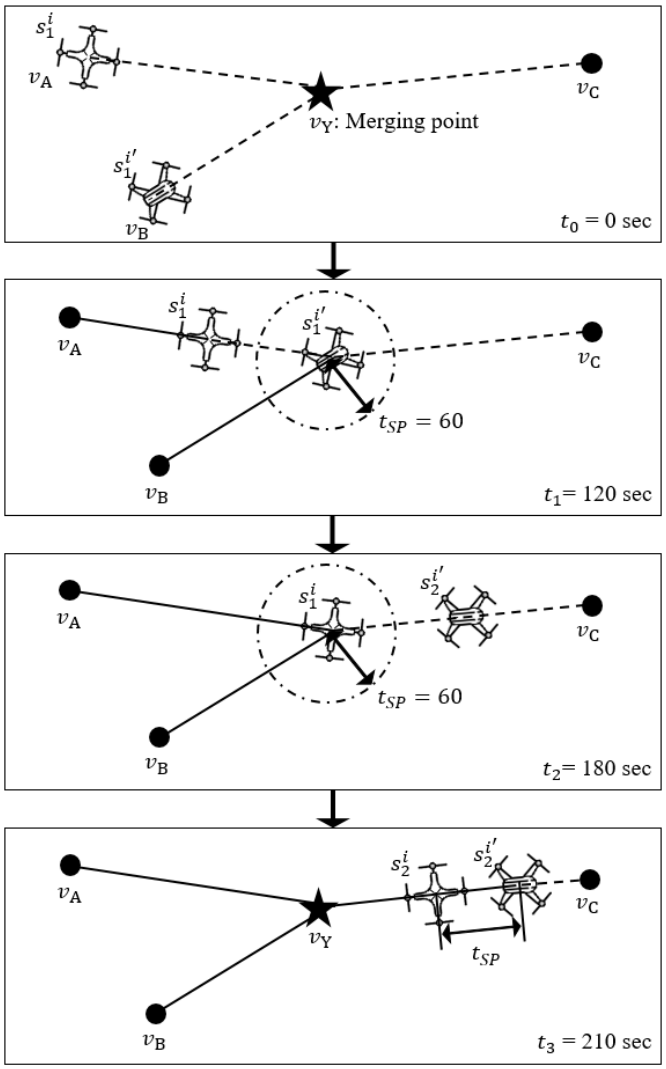


Figure 4.2: Time-based separation concept for merging points (homogeneous flights α_i and $\alpha_{i'}$ fly from their origin v_A and v_B to the same destination v_C via the merging point v_Y at the speed $\{s_1^i, s_2^i\}$ and $\{s_1^{i'}, s_2^{i'}\}$, respectively. The superscript and subscript of s are the flight index and the segment index, respectively)

For ease of understanding, we describe the concept with an example. At the time t_0 of Figure 4.2, α_i and $\alpha_{i'}$ fly toward the v_C through the same merging point, v_Y, at the speeds s_1^i and $s_1^{i'}$, respectively. The time t_1 when $\alpha_{i'}$ just passes through v_Y at the speed of s_1^i is stored at the merging point $T(v_Y)$, and no sUAS can pass through this point for t_{SP} seconds before and after t_1 where the time t_1 stands for the Separation

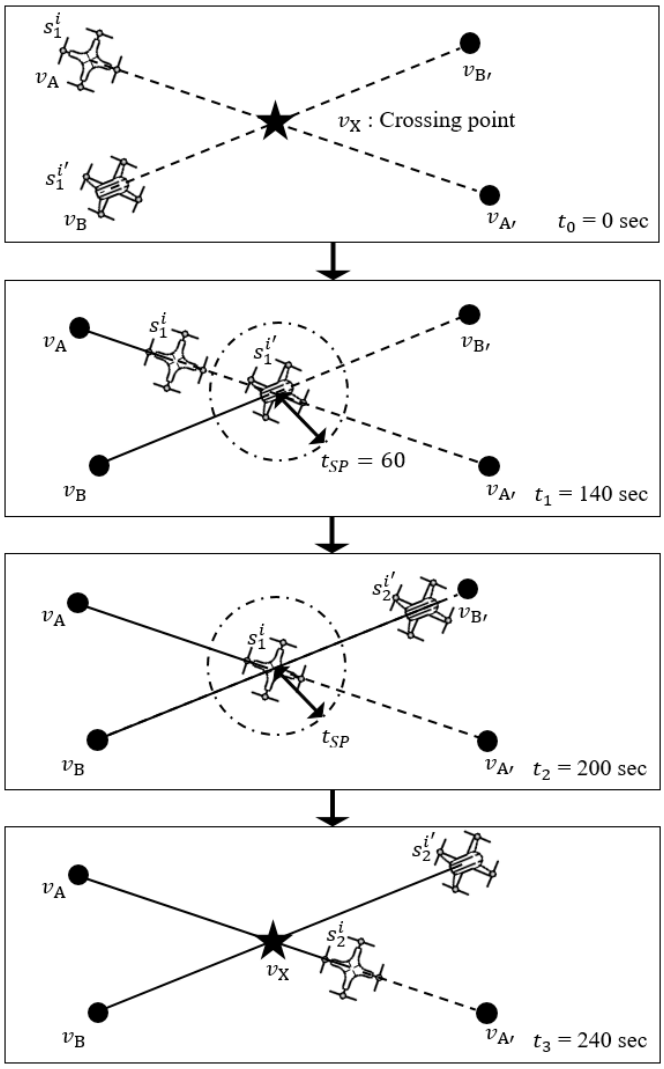


Figure 4.3: Time-based separation concept for crossing points (homogeneous flights α_i and $\alpha_{i'}$ fly from their origin v_A and v_B to their destinations $v_{A'}$ and $v_{B'}$ via the crossing point v_X at the speed $\{s_1^i, s_2^i\}$ and $\{s_1^{i'}, s_2^{i'}\}$, respectively. The superscript and subscript of s are the flight index and the segment index, respectively)

Assured Estimated Time of Arrival (SAETA). The SAETA t_2 is stored in $T(v_Y)$ in the same way. Then, α_i and $\alpha_{i'}$ traverse toward v_C at the speeds of s_2^i and s_2^i, respectively. If α_i and $\alpha_{i'}$ fly from v_Y to v_C at the speeds of s_2^i and $s_2^{i'}$, respectively, the separation between two sUASs will always be met the separation requirement or greater than that on the segment between v_Y and v_C. At the time t_3 of Figure 4.2, α_i and $\alpha_{i'}$ fly at the

same speed while maintaining the minimum separation. The time data at the waypoint v_{Y} is as follows:

$$T(v_{\mathrm{A}}) = \{0s\},$$
$$T(v_{\mathrm{B}}) = \{0s\},$$
$$T(v_{\mathrm{Y}}) = \{120s, 180s\},$$
$$T(v_{\mathrm{C}}) = \emptyset.$$

The time data at each waypoint obtained is used to calculate the optimisation problem that will be described in Section.4.3.

4.2.2 Mathematical Modelling

For the operational purposes, the route network-based operation is one of the concepts for the UTM system [17]. Thus, we model the high-density VLL airspace for a UTM system as a directed graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$, called an airspace graph. In the airspace graph, each vertex $v \in \mathcal{V}$ is a waypoint candidate to be traversed through Euclidean space of dimension two or three. Each edge $e \in \mathcal{E}$, corresponding a rectifiable curve, is a segment between some pair of waypoints in the airspace graph. In this model, a feasible flight route of sUAS α_i in the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$ is defined as follows:

Definition 10 As given in the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$, sUAS α_i to be routed in $\mathcal{G}$, and the initial waypoint $v_1^i \in \mathcal{V}$ and the final waypoint $v_{n_i}^i \in \mathcal{V}$, a flight route denoted by $p(\alpha_i)$ in $\mathcal{G} = (\mathcal{E}, \mathcal{V})$ is defined by a sequence of waypoints.

Thus, in the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$, there can be an abundance of flight route candidates denoted by $\mathcal{C}$ that satisfy the conditions as described in Definition 10. Through Definition 10, each flight route candidate $p(\alpha_i) \in \mathcal{C}$ can be given a corresponding flight route (a set of waypoints) as follows:

$$p(\alpha_i) : v_1^i, v_2^i, \ldots, v_{n_i}^i.$$

For each flight route $p(\alpha_i)$, there is a set of segments connecting waypoints, which is as follows:

$$\mathcal{E}(p(\alpha_i)) : e_{(1,2)}^i, e_{(2,3)}^i, \ldots, e_{(n_i-1,n_i)}^i.$$

Through this formulation, for each flight route candidate, it is assumed that a performance index for a set of segments $\mathcal{E}(p(\alpha_i))$ can be quantified as a set of positive numeric weight values, as follows:

$$\mathcal{W}(p(\alpha_i)) : w_1^i, w_2^i, \ldots, w_{n^i-1}^i.$$

Then, the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$ is transformed into a weighted directed graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W})$ by assigning a weight to each segment for all flight paths. In the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W})$, each flight route can be evaluated by summing all weights of $\mathcal{W}(p(\alpha_i))$, as follows:

$$\mathcal{T}(p(\alpha_i)) = \sum_{i=1}^{n_i-1} w_j^i. \tag{4.1}$$

Based on the airspace graph, $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W})$, the routing problem that minimises a performance index can be defined as follows:

Definition 11 Given an airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W})$ and corresponding all flight route candidates $\mathcal{C}$, the routing problem is defined as finding a flight route (or a sequence of waypoints) such that

$$p^*(\alpha_i) = \underset{p(\alpha_i) \in \mathcal{C}}{\operatorname{argmin}} \mathcal{T}(p(\alpha_i)). \tag{4.2}$$

The optimal flight route $p^*(\alpha_i)$ can be found by using the well-known shortest path algorithms such as Dijkstra's algorithm, A* algorithm, the brute-force search [18–20]. Although the optimal flight route $p^*(\alpha_i)$ can be obtained according to Definition 11,

the optimality of the flight route might be disturbed in the following stages: scheduling stage and conflict resolution stage to satisfy the separation requirement. Such a sequential approach can not only cancel optimality but also require additional works for scheduling, and sequencing and spacing.

Our objective of solving the flight planning problem is to maximise airspace capacity, which can be achieved by minimising the flight time of each sUAS. We also intend to minimise the disturbances in the sequential approaches by solving the routing, scheduling, and sequencing and spacing problems simultaneously. Our main idea for achieving the objective is to assign a weight to each segment (edge) of the airspace graph where the weight is expressed as sUAS' speed or the flight time. Therefore, each sUAS' speed profile also can be obtained by finding the optimal flight route in the airspace graph. We set the flight time of each sUAS as a weight $w \in \mathcal{W}$ to each segment $e \in \mathcal{E}$ of the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W})$. Note that, the flight time and speed of each sUAS are mutually interchangeable using the geographic data in the airspace graph. Each sUAS' speed profile is determined within its feasible speed range while satisfying the separation requirement from planned sUASs.

Another issue we have addressed is to satisfy the separation requirement between each pair of sUASs. To assign the flight time that satisfies the separation requirement to each edge of the airspace graph, we need data of the planned sUASs' flight time at each waypoint as discussed in Section.4.2.1. The time data containing the SAETA at each waypoint is included in the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W}, T)$ and be used to calculate weights of the airspace graph. By finding the optimal flight route of the airspace graph $\mathcal{G}$, then, we can obtain the schedule of the sUAS while satisfying the separation requirement simultaneously. Each sUAS $\alpha_i \in \mathcal{A}^+$ might have different weights $\mathcal{W}_i$ because of different specifications such as each sUAS' feasible speed range or the separation requirement as well as its initial and final positions. Thus, each sUAS might have its unique airspace graph $\mathcal{G}_i$ as shown in Figure 4.4.

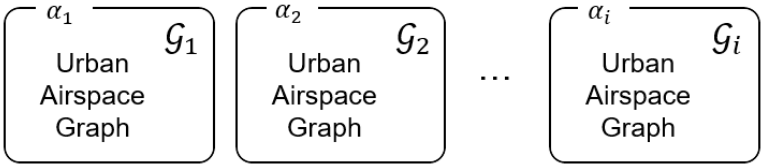


Figure 4.4: Unique urban airspace graphs for each flight

4.2.3 Calculating Flight Time Weights on $\mathcal{G}_i$ Suitable for Separation-compliant Speed Profiles

In this subsection, we propose a flight time weighting scheme to be applied to the airspace graphs $\mathcal{G}_i$ to fulfil the our objectives. An optimisation problem for calculating the weights is as follows:

$$\min \quad \sum_{j=1}^{n_i-1} \frac{d_j^i}{s_j^i} \tag{4.3}$$

$$\text{s.t.} \quad \frac{d_1^i}{s_1^i} + \frac{d_2^i}{s_2^i} + \cdots + \frac{d_{n_i-1}^i}{s_{n_i-1}^i} \geq \max T(v_{n_i}^i) + t_{SP} \tag{4.4}$$

$$\frac{d_1^i}{s_1^i} + \frac{d_2^i}{s_2^i} + \cdots + \frac{d_{n_i-2}^i}{s_{n_i-2}^i} \geq \max T(v_{n_i-1}^i) + t_{SP} \tag{4.5}$$

$$\vdots$$

$$\frac{d_1^i}{s_1^i} + \frac{d_2^i}{s_2^i} \geq \max T(v_3^i) + t_{SP} \tag{4.6}$$

$$\frac{d_1^i}{s_1^i} \geq \max T(v_2^i) + t_{SP} \tag{4.7}$$

$$s_{\text{min}}^i \leq s_1^i, s_2^i, \cdots, s_{n_i-1}^i \leq s_{\text{max}}^i \tag{4.8}$$

where s_{min}^i and s_{max}^i are minimum and maximum speed of flight α_i, respectively. Time t_{SP} is the minimum time-based separation requirement between sUASs at waypoints as shown in Figure 4.2. The objective function Equation (4.3) is to minimise the flight

time. The constraints from Equation (4.4) $\sim$ Equation (4.7) are for satisfying the minimum separation requirement at each waypoint. In Figure 4.1, for instance, if $\mathcal{E}(p(\alpha_i))$ consists of $n_i - 1$ segments for sUAS α_i, each segment requires the flight time that satisfies the separation requirement. Each of the left hand side of Equations (4.4) $\sim$ (4.7) is the flight time to the each segment, which must be greater than or equal to the time that satisfies the separation requirement. For the constraints, flight distances d between waypoints connected by segments and the SAETA T stored in $\mathcal{G}_i$ are required. The time data $T(v^i_{n_i})$ indicates the flight time that sUAS α_i fly over waypoint $v^i_{n_i}$. The time data is stored in T for each waypoint $v \in \mathcal{V}$, and T is updated every time sUAS is allocated and shared with $\forall \alpha_i \in \mathcal{A}^+$. Then, the optimisation problem we formulated can be solved by using any well-known linear programming algorithm. In this study, we utilise an interior point method. The weights can be obtained from the following equation:

$$\text{flight time} = \mathcal{W}_{i,t} = \frac{d(\alpha_i)}{s(\alpha_i)} \tag{4.9}$$

where $s(\alpha_i) : s^i_1, s^i_2, \cdots, s^i_{n_i-1}$ and $d(\alpha_i) : d^i_1, d^i_2, \cdots, d^i_{n_i-1}$ are a set of constant speeds and a set of flight distances, respectively.

Namely, in the optimisation problem, decision variables are constant speeds $s(\alpha_i)$, which are transformed into the flight time and assigned into flight time weights $\mathcal{W}_{i,t}$. Therefore, a solution of the airspace graph $\mathcal{G}_i = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}, T)$ can simultaneously provide a flight route and speed profile satisfying the minimum separation at all time. Flight time weights $\mathcal{W}_{i,t}$ can be derived from separation-compliant speed profile $s(\alpha_i)$ and segment distance $d(\alpha_i)$, $\forall \alpha_i \in \mathcal{A}^+$.

We also construct flight distance weights $\mathcal{W}_d$ on $\mathcal{G}_i$ to reflect the flight distance d as a second criterion. The flight distance weights are necessary to prioritise for multiple sUASs' flight planning problem when two or more sUASs arrive at the waypoint at the same time and determining the departure sequence, more details about the priority are in Section.4.3. The single sUAS flight planning problem that motivated this study can be formulated as follows:

Problem 3 Given an airspace graph $\mathcal{G}_i = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}, \mathcal{W}_d, T)$, sUAS α_i to be planned

in $\mathcal{G}_i$, and its origin $v_1^i \in \mathcal{V}$ and destination $v_{n_i}^i \in \mathcal{V}$ reachable from the origin, construct a flight route $p(\alpha_i)$ and a speed profile $s(\alpha_i)$, $\forall \alpha_i \in \mathcal{A}^+$ such that

- the separation requirement is satisfied from the planned sUASs,
- the speed profile for each sUAS must be within its feasible speed range,
- the airspace graphs are updated every time the route for a sUAS $\alpha_i \in \mathcal{A}^+$ is planned.

4.3 Flight Planning Algorithm for Multiple sUASs

Section 4.2 focuses on the single sUAS flight planning problem (inner loop) which can be formulated by assigning the proposed separation satisfied flight time weights to the airspace graph. By doing so, we can consider practical factors such as feasible speed ranges of the sUASs, and different separation requirements. In this study, we solve Problem 3 using Dijkstra's algorithm. However, the flight time weights of each sUAS only satisfy the separation requirement from planned sUASs.

In this section, we describe the outer loop of our algorithm which determines the order of arrival and departure of multiple sUASs. At each iteration, each unplanned sUAS finds its flight plan by solving the Problem 3 (inner loop) and one of them is allocated using the First Come First Served (FCFS) algorithm or the Last Come First Served (LCFS) algorithm (outer loop). The outer loop algorithm plays a important role to increase the route network-based urban airspace capacity. Through the case studies, we will show the difference between the FCFS algorithm and the LCFS algorithm according to the operational type.

Once inputs of the algorithm are given, the algorithm runs the flight planning process of the sUASs until $\mathcal{A}^+$ is empty (Line 3). The algorithm first generates the airspace graphs $\mathcal{G}_i^1, \forall \alpha \in \mathcal{A}^+$ (Line 2). Then, each sUAS has its unique airspace graph that contains flight time weights $\mathcal{W}_{i,t}^1$ determining the optimal flight route and its flight time. In order to reduce the complexity of the process that calculates the weights, the algorithm selects the h-shortest routes for $\mathcal{G}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_d, T^k), \forall \alpha_i \in \mathcal{A}^+$ using Yen's algorithm,

and generates $\bar{\mathcal{G}}_i^k$ only considering the h-shortest routes (Line 4-5) [21]. The optimal flight route $p^*(\alpha_i)$ is obtained by using Dijkstra's algorithm, $\forall \alpha_i \in \mathcal{A}^+$ (Line 6) [18]. The sUAS α_i^* is obtained by using the FCFS algorithm or the LCFS algorithm (Line 7), and allocated (Line 15). The sUAS α_i^* is removed from $\mathcal{A}^+$, and stored in $\mathcal{A}^-$ with its flight route $p^*(\alpha_i)$ and speed profile $s^*(\alpha_i)$ (Line 16). In the case of that more than two sUASs are arriving to the same destination (Line 8), the flight distance-based weights $\mathcal{W}_d$ are utilised to allocate a sUAS amongst them (Line 9-13). In any case, a sUAS is allocated and its flight data is shared with sUAS α_i, $\forall \alpha_i \in \mathcal{A}^+$. Based on the data each airspace graph $\mathcal{G}_i^k$ is updated, $\forall \alpha_i \in \mathcal{A}$ (Line 20).

We illustrate an example in Figure 4.5 for a better understanding of the outer loop concept of the proposed algorithm with the FCFS algorithm as the outer loop algorithm. In the first iteration, each sUAS generates its airspace graph $\mathcal{G}_i^1$ and $\bar{\mathcal{G}}_i^1$ is found, $\forall \alpha_i \in \mathcal{A}^+$. In the inner loop process, each sUAS's $p^*(\alpha_i)$ and $s^*(\alpha_i)$ is determined. The first arrival sUAS α_2 is planned using the FCFS algorithm. In the second iteration, sUAS α_A is planned through the same process as the first iteration. In the $(A)^{th}$ iteration, sUAS α_1 is planned last. Although in the example above we assume that the first come sUAS has the priority to be planned, the criteria or the outer loop algorithm to determine the priority can be changed.

Advantages of the proposed algorithm are as follows: (a) considering both the flight route and the flight speed to enlarge the solution searching space; (b) providing separation-compliant speed profiles that satisfy each sUAS's performance; (c) applicable for various types of route networks; (d) applicable for airspace capacity estimation.

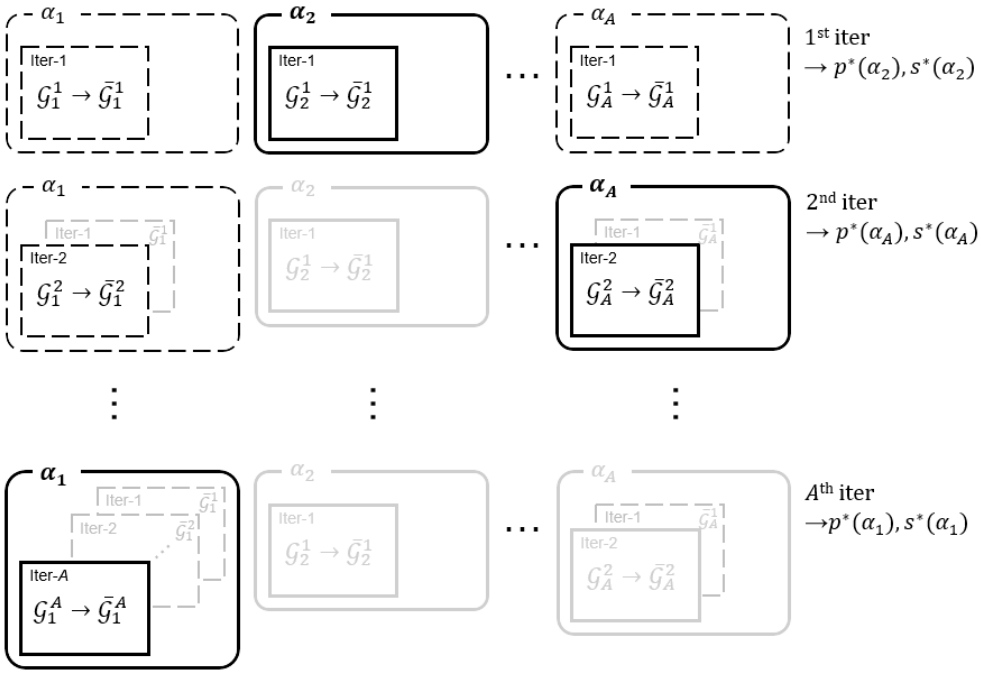


Figure 4.5: Iteratively generated airspace graph concept for multiple sUASs in Algorithm 3

Algorithm 3: Flight planning algorithm for multiple sUASs

Input: $v_1^i, v_{n_i}^i \in \mathcal{V}$, $\forall \alpha_i \in \mathcal{A}^+$, airspace information

Output: $p^*(\alpha_i), s^*(\alpha_i)$, $\forall \alpha_i \in \mathcal{A}^-$

$k = 1$

generate $\mathcal{G}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}^k, \mathcal{W}_d, T^k)$, $\forall \alpha_i \in \mathcal{A}^+$

while $\mathcal{A}^+ \neq \emptyset$ **do**

  find h-shortest routes for $\mathcal{G}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_d, T^k)$, $\forall \alpha_i \in \mathcal{A}^+$ (using Yen's algorithm);

  generate $\bar{\mathcal{G}}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}^k, \mathcal{W}_d, T^k)$, $\forall \alpha_i \in \mathcal{A}^+$ that only considers the best h routes, $\forall \alpha_i \in \mathcal{A}^+$;

  find $p^*(\alpha_i)$ of $\bar{\mathcal{G}}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}^k, T^k)$, $\forall \alpha_i \in \mathcal{A}^+$, $\forall \alpha_i \in \mathcal{A}^+$ (using Dijkstra's algorithm);

  $\alpha_i^* \leftarrow \underset{\forall \alpha_i \in \mathcal{A}^+}{\operatorname{argmin}} p^*(\alpha_i)$ (using the FCFS algorithm or the LCFS algorithm);

  if *There are more than two* α_i^* *exist* **then**

    find $p^*(\alpha_i)$ of $\bar{\mathcal{G}}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_d, T^k)$ amongst them (using Dijkstra's algorithm);

    $\alpha_i^\dagger \leftarrow \underset{\forall \alpha_i^*}{\operatorname{argmin}} p^*(\alpha_i)$ (using the FCFS algorithm or the LCFS algorithm) ;

    allocate $\alpha_i^\dagger$;

    remove $\alpha_i^\dagger$ from $\mathcal{A}^+$ and store $\alpha_i^\dagger$ in $\mathcal{A}^-$ with $p^*(\alpha_i)$ and $s^*(\alpha_i)$;

    share $\mathcal{A}^-$ with $\forall \alpha_i \in \mathcal{A}^+$;

  else

    allocate α_i^*;

    remove α_i^* from $\mathcal{A}^+$ and store α_i^* in $\mathcal{A}^-$ with $p^*(\alpha_i)$ and $s^*(\alpha_i)$;

    share $\mathcal{A}^-$ with $\forall \alpha_i \in \mathcal{A}^+$;

  end

  $k = k + 1$;

  update $\mathcal{G}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}^k, \mathcal{W}_d, T^k)$, $\forall \alpha_i \in \mathcal{A}^+$;

end

4.4 Numerical Simulations

Last-mile delivery and first-mile delivery have been chosen to demonstrate the proposed algorithm. Scenarios are based on a town 'Oldbrook' in Milton Keynes, which is one of the well-planned cities in United Kingdom as shown in Figure 4.6. Two-dimensional infrastructure data of the town is obtained from Google Earth Pro. Both scenarios have an area size of 0.98 km^2. For this two operation cases, we construct two route networks over roads which are similar to the en-route airspace for commercial aicraft. In this route networks, two layers of nodes are set above the roads at the height of 25 meters and 30 meters for eastbound and westbound as shown in Figure 4.7 and Figure 4.8, respectively. Each route network consists of 68 points (1 retail point (red pin) and 67 service points (green pins)) as shown in Figure 4.6, and 103 directed routes as shown in Figure 4.7 and Figure 4.8. Thirty sUASs traverse from each sUAS' start point to each sUAS' final point. Both cases consider only one type of the sUAS. The sUAS type can be changed based on thier missions and operators. Also, it is assumed that the separation requirement including landing and take-off during the flight between sUASs is 5 seconds as shown in Table 4.1.

Table 4.1: sUAS specification & separation requirements

Permissible Speed Range [km/h]	[5 - 25]
Horizontal Separation [sec]	5
Departure Separation [sec]	5
Arrival Separation [sec]	5

The outer loop of Algorithm 3 to allocate sUASs' departure and arrival order is one of the factors that determine the urban airspace capacity, throughput, etc. For the outer loop we present two simulations with both the FCFS algorithm and the LCFS algorithm for both the last-mile delivery case (1-to-$\mathcal{M}$) and the first-mile delivery ($\mathcal{N}$-to-1) case to demonstrate the effect of the outer loop algorithm as well as the efficiency of Algorithm 3.

Figure 4.6: Oldbrook, Milton Keynes, United Kingdom (*Google Earth, 2018*)

4.4.1 Case Study: Last-mile Delivery (1-to-$\mathcal{M}$)

For the last mile delivery both the FCFS algorithm and the LCFS algorithm to allocates sUASs' departure sequence are applied (for the latter case Line 7 and Line 10 of Algorithm 3 are replaced to the LCFS algorithm). Table 4.2 and 4.3 show each sUAS' flight details when the FCFS algorithm and the LCFS algorithm are applied, respectively. Both of the outer loop algorithms generate 30 sUASs' flight paths and its schedules while satisfying the minimum separation requirements. Additionally, from comparison results between both of the algorithms as shown in Table 4.4 and Figure 4.9, we can see that the mission completion time of the LCFS algorithm case is 34.5% faster than the FCFS algorithm case. In the latter case, also, each sUASs' mission completion time is more evenly distributed than the FCFS algorithm.

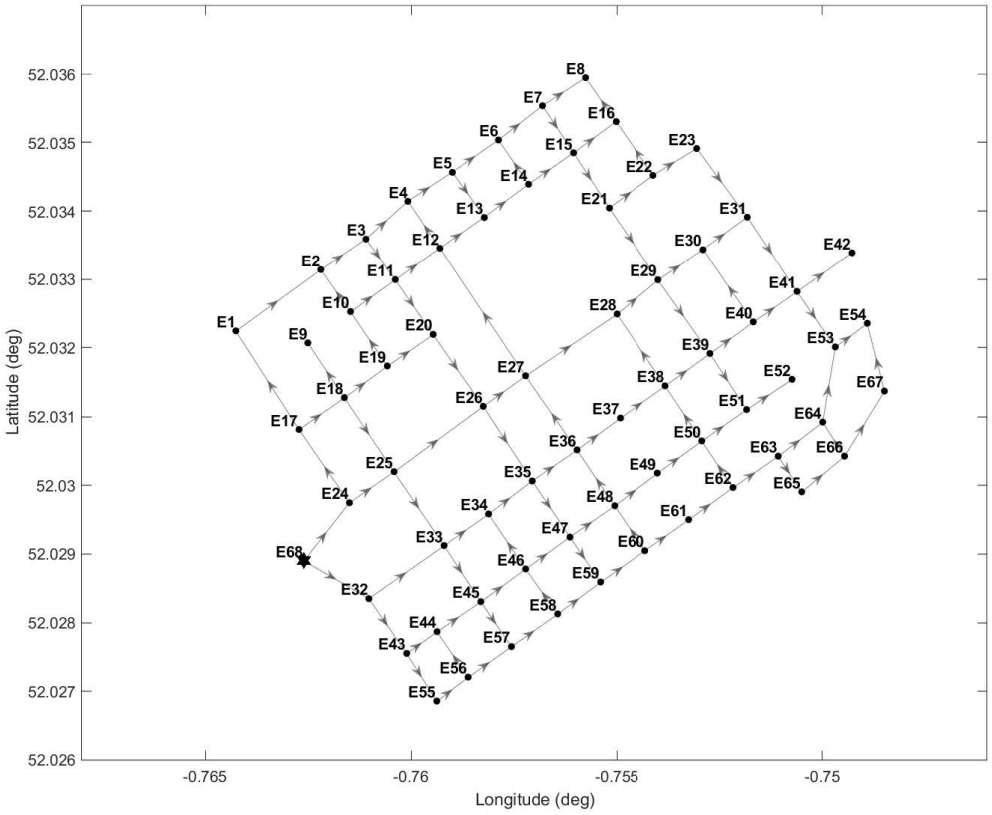


Figure 4.7: Eastbound, 1 retail point (E68) and 67 service points (E1 $\sim$ E67) and 103 directed routes

4.4.2 Case Study: First-mile Delivery ($\mathcal{N}$-to-1)

We consider the worse case of the first mile delivery, which is that every sUAS departs at the same time. For this case, both the FCFS algorithm and the LCFS algorithm to allocates sUASs' departure sequence are applied (for the latter case Line 7 and Line 10 of Algorithm 3 are replaced to the LCFS algorithm). Note that, due to various factors such as the number of sUASs, feasible speed ranges, and minimum separation requirements, Algorithm 3 may not be able to generate flight plans for the sUAVs satisfying the minimum separation requirements for the $\mathcal{N}$-to-1 case. In this worst case of the first mile delivery, particularly, Algorithm 3 with the LCFS algorithm for the outer loop can-

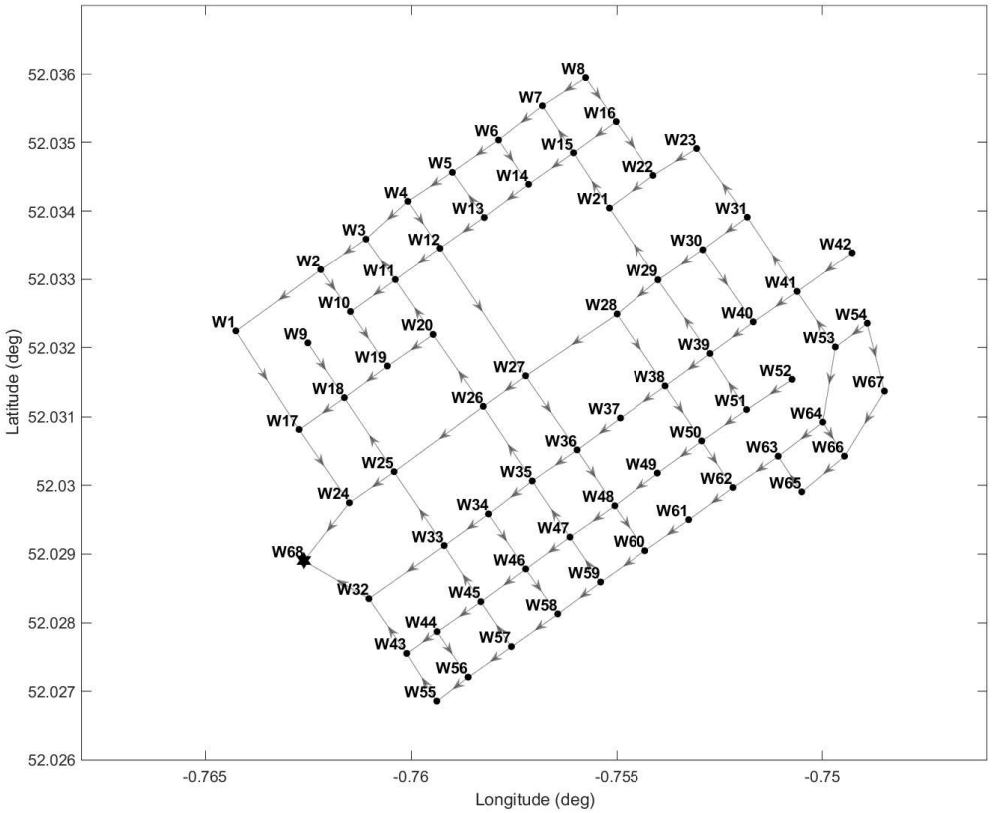


Figure 4.8: Westbound, 1 retail point (W68) and 67 service points (W1 $\sim$ W67) and 103 directed routes

not generate separation assured flight plans while the FCFS algorithm generates each sUAS' flight plan that satisfies the minimum separation requirement. Table 4.5 shows each sUAS' flight details when the FCFS outer loop algorithm is applied for the given start point and final point of each sUAS.

The computation time of the algorithm for this example on a Windows 10 OS 3.4 GHz desktop computer with 16 GB RAM is shown in Table 4.6. The computation time will be significantly decreased if we solve each sUAS' inner loop in a decentralised way, which is assumed as a centralised manner in this study.

Table 4.2: Flight planning results for 30 flights for the last-mile delivery (Outer loop: FCFS algorithm)

Flight ID	**Waypoint sequence (arrival time at each waypoint [sec]) speed profile [km/h]**
⋮	⋮
α_4	E68(180) - E24(197) - E17(217) - E1(244) - E2(268) - E3(280) - E4(293) 25 - 25 - 25 - 25 - 25 - 25
α_5	E68(220) - E24(237) - E17(257) - E1(284) - E2(308) - E3(320) - E4(333) - E5(345) 25 - 25 - 25 - 25 - 25 - 25 - 25
⋮	⋮
α_{22}	E68(130) - E32(147) - E33(169) - E34(181) - E35(193) - E36(206) 25 - 25 - 25 - 25 - 25
α_{23}	E68(170) - E32(188) - E33(209) - E34(221) - E35(234) - E36(246) - E37(258) - E38(271) 25 - 25 - 25 - 25 - 25 - 25 - 25
⋮	⋮

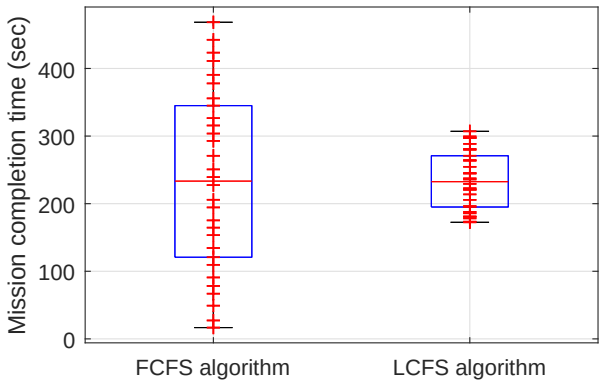


Figure 4.9: Standard deviation comparison of mission completion time between the FCFS algorithm and the LCFS algorithm for the outer loop of Algorithm 3

Table 4.3: Flight planning results for 30 flights for the last-mile delivery (Outer loop: LCFS algorithm)

Flight ID	**Waypoint sequence (arrival time at each waypoint [sec]) speed profile [km/h]**
⋮	⋮
α_4	E68(110) - E24(127) - E17(147) - E1(173) - E2(197) - E3(210) - E4(223) 25 - 25 - 25 - 25 - 25 - 25
α_5	E68(70) - E24(87) - E17(107) - E1(133) - E2(157) - E3(170) - E4(183) - E5(195) 25 - 25 - 25 - 25 - 25 - 25 - 25
⋮	⋮
α_{22}	E68(160) - E32(177) - E33(198) - E34(211) - E35(223) - E36(236) 25 - 25 - 25 - 25 - 25
α_{23}	E68(120) - E32(137) - E33(158) - E34(171) - E35(183) - E36(196) - E37(208) - E38(221) 25 - 25 - 25 - 25 - 25 - 25 - 25
⋮	⋮

Table 4.4: Comparison between the FCFS algorithm and the LCFS algorithm for the last-mile delivery

	FCFS algorithm	**LCFS algorithm**
Total flight time [sec]	7035.1	7031.1
Total flight distance [km]	19.3	19.3
Mission completion time [sec]	468.7	307.0

Table 4.5: Flight planning results for 30 flights for the first-mile delivery (Outer loop: FCFS algorithm)

Flight ID	**Waypoint sequence (arrival time at each waypoint [sec]) speed profile [km/h]**
⋮	⋮
α_4	W4(0) - W3(13) - W2(25) - W1(49) - W17(76) - W24(96) - W68(113) 25 - 25 - 25 - 25 - 25 - 25
α_5	W5(0) - W4(13) - W3(26) - W2(39) - W10(50) - W19(65) - W18(77) - W17(91) - W24(111) - W68(128) 24 - 25 - 25 - 25 - 25 - 25 - 24 - 25 - 25
⋮	⋮
α_{22}	W36(0) - W35(13) - W34(25) - E33(39) - W32(60) - W68(77) 25 - 25 - 24 - 25 - 25
α_{23}	W38(0) - W37(15) - W36(28) - E35(41) - W34(56) - E33(69) - W32(90) - W68(107) 22 - 24 - 24 - 22 - 25 - 25 - 25
⋮	⋮

Table 4.6: Computational time for the cases [sec]

Last mile delivery with FCFS algorithm	19.8
Last mile delivery with LCFS algorithm	12.1
First mile delivery with FCFS algorithm	21.8

4.5 Conclusions

In this study, the flight planning algorithm for multiple sUASs is proposed with the route network over roads. The algorithm generates routes and schedules of the sUASs to minimise each sUAS' flight time in the given route network. The resulting flight plans satisfy the minimum separation requirement between the sUASs at all times. The algorithm searches each sUAS' flight route and its speed profile simultaneously, which means the flight plans are found in a large solution space. Also, each sUAS' flight planning can be done in parallel, which may significantly reduce the computation time. Through the last mile delivery case and the first mile delivery case, we show that the practical and detailed flight plans, which can support to analyse the route network urban airspace and to establish requirements for safe operation. This algorithm positively will benefit the backbone of designing the concept of the UTM system to be integrated into the ATM system. However, additional dynamic constraints are required to improve the algorithm.

To make the proposed algorithm a practical tool for flight planning for multiple sUASs, more research is required to considering factors such as determining flight order, hovering manoeuvre in a highly congested airspace, uncertainties such as wind effect, dynamical behaviour of the sUAS, etc.

1. *Outer loop algorithm*: Determining departure and arrival sequences which is the outer loop of our algorithm is an important factor for both flight planning and airspace capacity. Although we applied the centralised FCFS algorithm and the LCFS algorithm to decide the sequence of arrival and departure, there are many different algorithms that use many different approaches and techniques. We expect that the algorithm can be changed according to the airspace structure and the use of airspace.

2. *$\mathcal{N}$-to-$\mathcal{M}$ case*: There could be many airspace users who have different retail points with different priorities. Therefore, the algorithm should consider multiple start points, and each airspace users' flight plans also should be calculated separately. Here, determining the multiple airspace users' arrival and departure

order is also a key issue to be addressed to efficiently manage and operate the airspace. There can be several approaches to tackle the issue such as intention-based approaches, criteria-based approaches, etc.

3. *Uncertainty*: One possible approach to include wind effect is to increase the minimum separation requirement, although the approach will negatively affect the airspace capacity and throughput. If the static wind effect could be calculated as a vector, the algorithm can directly consider the effect. Also, contingency planning will be available with our algorithm if well-structured emergency infrastructures exist such as emergency landing points.

References

[1] Amazon.com Inc., "Determining Safe Access with a Best- Equipped, Best-Served Model for Small Unmanned Aircraft Systems," *NASA UTM 2015: The Next Era of Aviation*, 2015.

[2] C. A. Thiels, J. M. Aho, S. P. Zietlow, and D. H. Jenkins, "Use of unmanned aerial vehicles for medical product transport," *Air Medical Journal*, vol. 34, no. 2, pp. 104–108, 2015.

[3] Federal Aviation Administration, "Operation and Certification of Small Unmanned Aircraft Systems," Federal Aviation Administration, Tech. Rep. 124, 2016. [Online]. Available: https://federalregister.gov/a/2016-15079

[4] Geister and Dagi, "Concept for Urban Airspace Integration DLR U-Space Blueprint," Institute of Flight Guidance, Tech. Rep. December, 2017. [Online]. Available: http://www.dlr.de/fl/desktopdefault.aspx/tabid-11763/20624{_}read-48305/

[5] J. Hoekstra, S. Kern, O. Schneider, F. Knabe, and B. Lamiscarre, "Metropolis – Urban Airspace Design," Technical University of Delft National, Tech. Rep., 2015.

[6] J. Rios, D. Mulfinger, J. Homola, and P. Venkatesan, "NASA UAS traffic management national campaign: Operations across Six UAS Test Sites," in *2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)*. IEEE, sep 2016, pp. 1–6.

[7] M. F. B. Mohamed Salleh and K. H. Low, "Concept of Operations (ConOps) for Traffic Management of Unmanned Aircraft Systems (TM-UAS) in Urban Environment," in *AIAA Information Systems-AIAA Infotech @ Aerospace*, no. January. Reston, Virginia: American Institute of Aeronautics and Astronautics, jan 2017, pp. 1–13.

[8] S. Bortoff, "Path planning for UAVs," *Proceedings of the 2000 American Control Conference. ACC (IEEE Cat. No.00CH36334)*, no. June, pp. 364–368 vol.1, 2000.

[9] J. Enright, E. Frazzoli, K. Savla, and F. Bullo, "On Multiple UAV Routing with Stochastic Targets: Performance Bounds and Algorithms," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, no. August. Reston, Virigina: American Institute of Aeronautics and Astronautics, aug 2005, p. 5830.

[10] G. B. Lamont, J. N. Slear, and K. Melendez, "UAV swarm mission planning and routing using multi-objective evolutionary algorithms," *IEEE Symposium Computational Intelligence in Multicriteria Decision Making*, no. Mcdm, pp. 10–20, 2007.

[11] S. G. Manyam, S. Rasmussen, D. W. Casbeer, K. Kalyanam, and S. Manickam, "Multi-UAV routing for persistent intelligence surveillance & reconnaissance missions," *2017 International Conference on Unmanned Aircraft Systems, ICUAS 2017*, pp. 573–580, 2017.

[12] K. Yu, A. K. Budhiraja, and P. Tokekar, "Algorithms for Routing of Unmanned Aerial Vehicles with Mobile Recharging Stations," *IEEE International Conference on Robotics and Automation 2018*, pp. 1–5, mar 2017.

[13] K. Dorling, J. Heinrichs, G. G. Messier, and S. Magierowski, "Vehicle Routing Problems for Drone Delivery," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 1, pp. 70–85, 2017.

[14] M. Samà, A. D'Ariano, P. D'Ariano, and D. Pacciarelli, "Scheduling models for optimal aircraft traffic control at busy airports: Tardiness, priorities, equity and violations considerations," *Omega (United Kingdom)*, vol. 67, pp. 81–98, 2017.

[15] S. Bae, H.-S. Shin, C.-H. Lee, and A. Tsourdos, "A New Multiple Flights Routing and Scheduling Algorithm in Terminal Manoeuvring Area," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*. IEEE, sep 2018, pp. 1–9.

[16] S. Bae, H.-S. Shin, and A. Tsourdos, "A New Graph-Based Flight Planning Algorithm for Unmanned Aircraft System Traffic Management," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*. IEEE, sep 2018, pp. 1–9.

[17] M. F. B. Mohamed Salleh, C. Wanchao, Z. Wang, S. Huang, D. Y. Tan, T. Huang, and K. H. Low, "Preliminary Concept of Adaptive Urban Airspace Management for Unmanned Aircraft Operations," in *2018 AIAA Information Systems-AIAA Infotech @ Aerospace*, no. January. Reston, Virginia: American Institute of Aeronautics and Astronautics, jan 2018, pp. 1–12.

[18] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, dec 1959.

[19] P. Hart, N. 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.

[20] R. T. Wong, "Combinatorial Optimization: Algorithms and Complexity (Christos H. Papadimitriou and Kenneth Steiglitz)," *SIAM Review*, vol. 25, no. 3, pp. 424–425, jul 1983.

[21] J. Y. Yen, "Finding the K Shortest Loopless Paths in a Network," *Management Science*, vol. 17, no. 11, pp. 712–716, jul 1971.

Chapter 5

Urban Airspace Capacity Analysis using Flight Planning Algorithms

In this chapter, we investigate the route network-based urban airspace capacity analysis based on the proposed algorithm (with modification) in Chapter 4 for four drone delivery operation types (1-to-$\mathcal{M}$, $\mathcal{M}$-to-1, $\mathcal{N}$-to-$\mathcal{M}$, $\mathcal{M}$-to-$\mathcal{N}$).

5.1 Introduction

Many civilian applications of small Unmanned Aircraft Systems (sUASs) have been envisioned to be operated in congested urban airspace for various purposes such as deliveries of medical supplies, deliveries of packages to rural areas [1–5]. Such applications are expected to significantly increase the quality of these services. In [6], authors compared between the ground vehicles and sUASs, and concluded that it is advantageous to use the sUASs for delivering packages, although the current sUASs technically cannot take heavy cargoes and many cargoes like vans. Today, however, there is no infrastructure to enable and safely manage the use of urban airspace and sUAS operations. Learning from the history of the Air Traffic Management (ATM) system, therefore, many organisations investigate concepts, functional designs, and prototypes of Unmanned Aircraft System (UAS) Traffic Management (UTM) system to support safe and efficient sUAS operations for many applications in urban airspace [6–11].

One of the most fundamental tasks facing the stakeholders of the UTM system over the world involves defining, measuring, and predicting of the capacity of urban airspace. The capacity can be analysed from various perspectives such as safety, performance efficiency, conflicts, noise, communication spectrum, etc. The difficulty of the task is due to the numerous factors that affect the urban airspace capacity. In the ATM system most similar to the UTM system, the general notion of the airspace capacity is the number of flights that can be accommodated in a given airspace within a given time. In the literature, authors proposed capacity estimation approaches, which are evaluated from workload of air traffic controllers and pilots [12–16]. These approaches are highly dependent on subjective assessment or judgement of the air traffic controllers. However, such human dependent airspace capacity estimation approaches are unsuitable for the UTM system, which is expected to operate a large number of sUASs by a small number of people or by itself.

In [17], recently, authors showed that the threshold based mathematical definition to estimate the capacity for free flight based low altitude airspace. The authors used two metrics to estimate the capacity: safety (Total Loss of Flight per Flight Hour) and performance (Percentage Extension of Travel Distance). In [18], authors investigated new deconfliction schemes for unmanned aerial vehicles in high-density Very Low Level (VLL) uncontrolled airspace, and assessed the single- and multi-layered airspace designs. Their aim is to provide a framework for choosing resolution strategies for regulators and policymakers for the UTM system. In [11], authors showed the determining the situation-appropriate route network for unmanned aircraft in the urban airspace. The authors suggested three types of route networks (AirMatrix, over Buildings, over Roads), and showed the results of the capacity and throughput for each route network. The results are obtained by generating five two-way routes for aerial delivery missions from the supply point to the five service points where the routing problem is solved with an assumption that once a sUAS occupies a segment, then any other sUASs cannot traverse the route. However, the approach causes very conservative analysis results and might disturb the optimality of solutions of following problems such as scheduling, separation management, although these have a computational advantage. As presented in the literature [9, 10, 17–19], on the development of advanced

technology, the free flight based operation in congested urban airspace can be an option for the UTM system. One of the other options is route network-based operations as described in [11, 20]. In the ATM context, also, opinions on the types of airspace operations diverge into two groups, a reduction of the constraints research [21, 22] and more structured operations preferred research [23, 24].

Inspired by [11,20,25,26], we investigate strategies for analysing high-density VLL structured and controlled urban airspace capacity, and the primary aim of this work is to support regulators, airspace designers, and policymakers for determining tailored strategies for considering regional characteristics and environment. In this study, we perform capacity analysis to assess four different drone delivery operation types (1-to-$\mathcal{M}$, $\mathcal{M}$-to-1, $\mathcal{N}$-to-$\mathcal{M}$, and $\mathcal{M}$-to-$\mathcal{N}$, where $\mathcal{M} > \mathcal{N}$) in a route network over the roads in a given urban airspace. In order to overcome the disadvantages of the five two-way route planning used in [11], we extend the flight planning algorithm proposed in our previous studies to perform quantitative capacity analysis [27,28]. The proposed algorithm that iteratively solves multiple sUASs' flight planning problems consists of an inner loop algorithm and an outer loop algorithm. At each iteration, the inner loop algorithm generates each sUAS' flight route and speed profile by solving the shortest path problem, and the outer loop algorithm determines departure and arrival sequences. To suggest an arrival and departure algorithm for each operation type, we utilise two different outer loop algorithms, which are the First Come First Served (FCFS) algorithm and the Last Come First Served (LCFS) algorithm, for allocating the sequences. We utilise obtained flight plans to analyse the urban airspace by defining four metrics as follows: total flight time; mission completion time; total flight distance; and normalised number of conflicts. For each operation type, we conduct 100 Monte Carlo simulations for both the FCFS algorithm and the LCFS algorithm. From the results, we can either find an appropriate outer loop algorithm for regional characteristics and the cases or reconfigure route networks as it is required.

The structure of this study is as follows. We first focus on a design of the route network over roads in Section 5.2. Section 5.3 provides details of the flight planning algorithms for the different cases, and describes the metrics to analyse the urban airspace capacity. In Section 5.4, we perform the MC simulations for each operation type and

analyse its results. In Section 5.5, we conclude and discuss the results.

5.2 Urban Airspace Structure Concept: Route Network Over Roads

Many studies have focused on the sUAS path planning problem [29–39]. More than half of them solve the problem with the unstructured airspace assumption. For urban airspace users in high-density VLL urban airspaces, however, many difficulties are expected with the sUAS studies, which are based on free routing or free flight operations, such as the risk due to unforeseen crash, invasion of privacy. On the other hand, the structured and controlled airspace can predict crashes more easily than the unstructured airspace, and privacy can be considered in an urban airspace structure design stage.

In this section, we suggest a route network-based urban airspace structure, which can be a potential candidate for the high-density VLL airspace. The proposed urban airspace structure concept resembles the airspace structure used in en-route phases of the ATM system today that can be considered as a combination of the Layers concept and the Tubes concept among four concepts Full Mix, Layers, Zones, and Tubes presented in [26]. The characteristics of the suggested airspace structure concept are as follows: consist of waypoints (nodes) and segments (edges) connecting some pair of waypoints; flight direction is strictly defined for all segments; stack at least two layers; separate traffic based on a time-based separation concept within each layer. As shown in Figure 5.1, for example, there is an eastbound layer and a westbound layer, and each layer consists of parallel aligned northbound segments and southbound segments. Such a route network over roads with directions in urban airspace will increase the predictability of traffic by using flight plans. In this concept, the urban airspace structure can be thought of as a directed graph (or digraph) $\mathcal{G} = (\mathcal{E}, \mathcal{V})$, called an *airspace graph* to highlight the aerospace context. In the airspace graph $\mathcal{G}$, node $v \in \mathcal{V}$ being a point in a Euclidean space of dimension two or three representing a waypoint candidate to be traversed. Edge $e \in \mathcal{E}$, corresponds a rectifiable curve, is a segment between some pair of waypoints in the airspace graph $\mathcal{G}$.

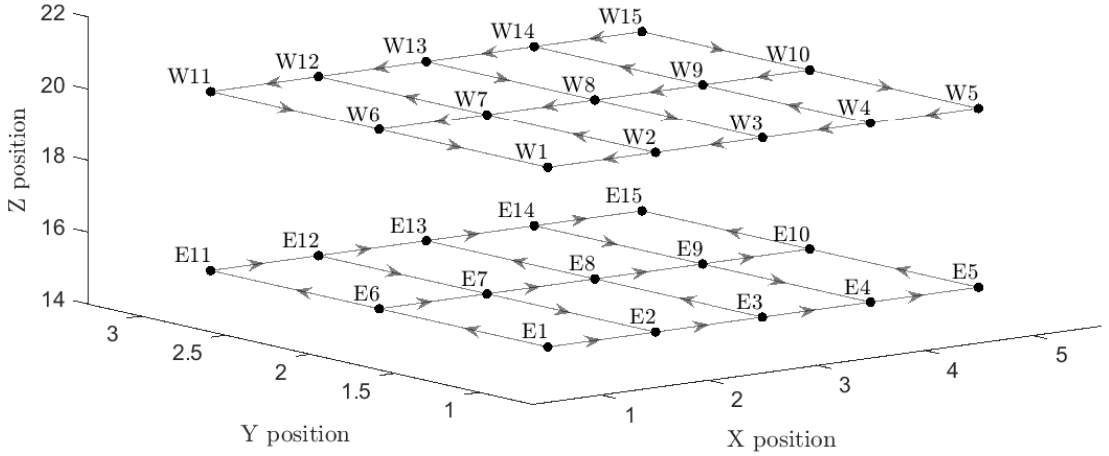


Figure 5.1: An example of the proposed route network-based urban airspace structure concept

Main assumptions we make for the route network-based operations in this study are as follows: each flight route is composed of a series of linear segments as shown in Figure 5.2; each sUAS flies along each segment at a constant speed; each sUAS is able to exactly follow the flight route without deviation; a time-based separation concept is utilised; uncertainties produced by external sources are neglected such as adverse weather; flight plans are shared between sUASs; the hovering and vertical manoeuvres are not considered; sUASs have limited payload and flight range; we only consider flights within each layer (e.g., no transitions between layers), but the transitions can be adopted by establishing the vertical segments.

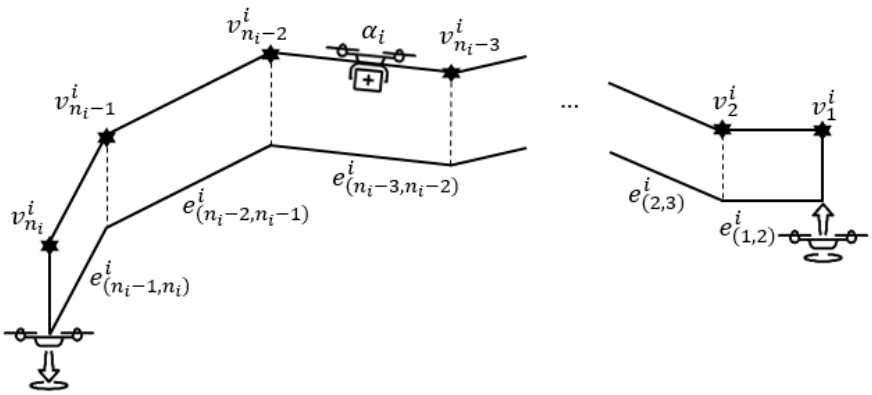


Figure 5.2: A series of linear segments (sUAS α_i's flight route from its origin v^i_1 to destination $v^i_{n_i}$)

5.3 Flight Planning Algorithms

In this section, we describe the flight planning algorithm that is used to analyse the route network-based urban airspace. The algorithm that solves multiple sUASs flight planning problems consists of the inner loop algorithm and the outer loop algorithm. The inner loop algorithm optimises each sUAS' flight plan that minimises its flight time while satisfying the time-based minimum separation requirement from planned sUASs. Then, the outer loop algorithm allocates arrival and departure sequences of the sUASs.

Definition 12 We define the flight planning problem as determining the following: (a) each sUAS' flight route, (b) its separation-compliant speed profile including arrival times at each waypoint, and (c) arrival and departure sequences (if it is a multiple sUASs flight planning problem).

This section consists of five parts: (a) a description of the time-based separation concept; (b) flight planning problem formulation for each sUAS (inner loop); (c) the entire flight planning algorithm with two different outer loop algorithms; (d) a flight planning example using the proposed algorithm for a last-mile delivery case (1-to-$\mathcal{M}$); (e) a description of four metrics to analyse the urban airspace capacity quantitatively.

5.3.1 Separation Concept in Route Network-Based Unban Airspace for Heterogeneous sUASs

Conflict detection is activated when separation of two sUASs is less than a minimum separation criterion. In this study, time-based separation is applied instead of the widely used distance-based separation to stabilise the spacing between the sUASs. In this concept, a sUAS will occupy a waypoint when the sUAS passes the waypoint for a predetermined time interval. Within this time interval, no sUAS is allowed to traverse through the waypoint to satisfy the separation requirement. For each waypoint, a time interval list is maintained to keep track of the times at which the waypoint is expected to be occupied. These lists, called *time data* in this study, are shared between all sUASs and updated whenever a sUAS is allocated in the urban airspace. By satisfying the separation assurance flight time at each waypoint rather than adjusting a distance between the sUASs, we can obtain results that satisfy the minimum separation along routes as well as at waypoints at all times.

The separation concept allows time-based separation to be satisfied at merging points and crossing points as shown in Figure 5.3 and Figure 5.4, respectively. At the time t_0 of Figure 5.3, sUAS α_i and sUAS $\alpha_{i'}$ fly toward the v_{C} through the same merging point v_{Y} at the speeds s_1^i and $s_1^{i'}$, respectively. The SAETA t_0 is stored at each start point $T(v_{\mathrm{A}})$ and $T(v_{\mathrm{B}})$, and no sUAS can pass through both of the waypoints for SeParation time $t_{SP}^{(i'',i)}$ seconds and $t_{SP}^{(i''',i)}$ seconds before and after the SAETA t_0, respectively. Note that, where $t_{SP}^{(i',i)}$ is depending on the types of α_i and $\alpha_{i'}$. The SAETA t_1 of Figure 5.3 when $\alpha_{i'}$ just passes through v_{Y} is stored at the merging point $T(v_{\mathrm{Y}})$, and no sUAS can pass through this waypoint for $t_{SP}^{(i',i)}$ seconds before and after t_1. The SAETA t_2 of Figure 5.3 is stored in $T(v_{\mathrm{Y}})$ in the same way. Then, α_i and $\alpha_{i'}$ traverse toward v_{C} at the speeds of s_2^i and $s_2^{i'}$, respectively. If α_i and $\alpha_{i'}$ fly from v_{M} to v_{C} at the speeds of s_2^i and $s_2^{i'}$, respectively, the separation between two sUASs will always be met the separation requirement or greater than that on the segment between v_{Y} and v_{C}. At the time t_3 of Figure 5.3, α_i and $\alpha_{i'}$ fly at the same speed while maintaining the

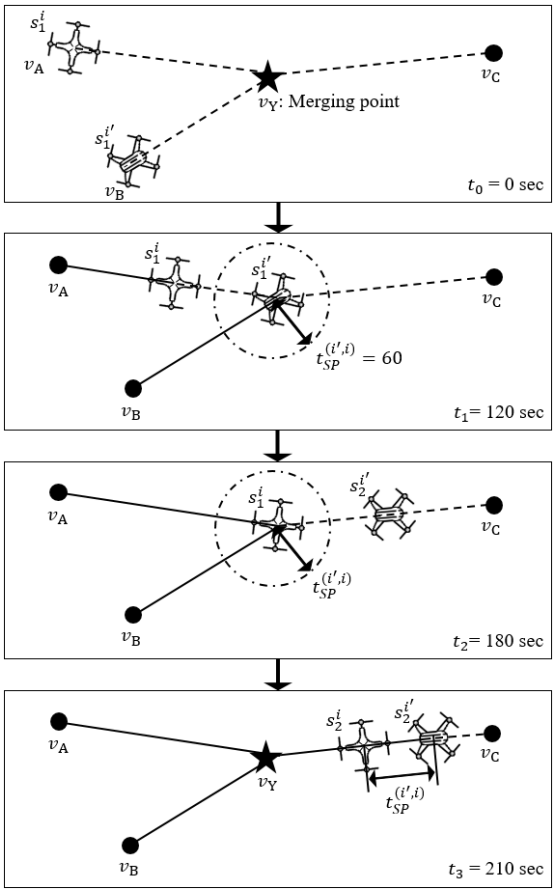


Figure 5.3: Time-based separation concept for merging points(α_i and $\alpha_{i'}$ fly from their origin v_{A} and v_{B} to v_{C} through the merging point v_{Y} at the speed $\{s_1^i, s_2^i\}$ and $\{s_1^{i'}, s_2^{i'}\}$, respectively. The superscript and subscript of s are the sUAS index and the segment index, respectively)

minimum separation. The time data from t_0 to t_3 at each waypoint is as follows:

$$T(v_{\mathrm{A}}) = \{0s\},$$

$$T(v_{\mathrm{B}}) = \{0s\},$$

$$T(v_{\mathrm{Y}}) = \{0s, 120s, 210s\},$$

$$T(v_{\mathrm{C}}) = \emptyset.$$

If no sUAS has passed waypoint v_{C}, then $T(v_{\mathrm{C}}) = \emptyset$. Such time data at each waypoint is used to formulate a flight planning problem that will be described in the following

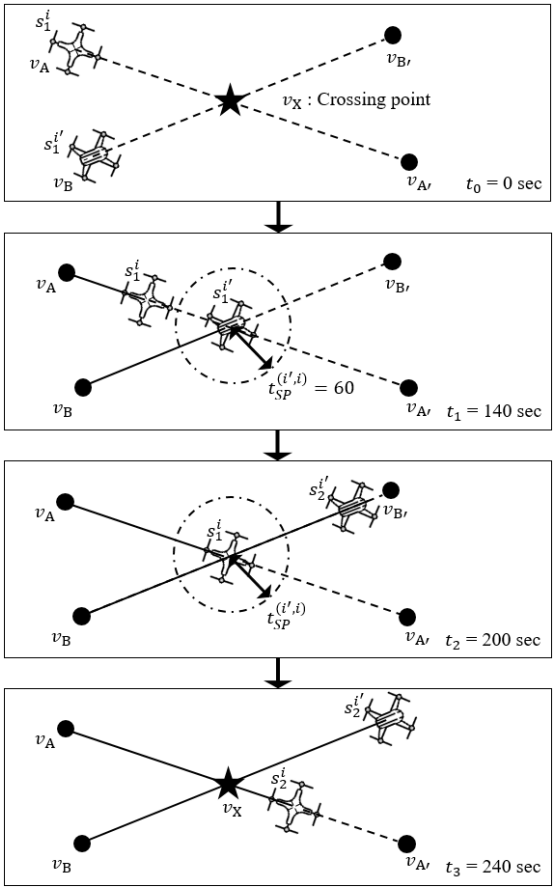


Figure 5.4: Time-based separation concept for crossing points (α_i and $\alpha_{i'}$ fly from their origin point v_{A} and v_{B} to their final point $v_{\mathrm{A}'}$ and $v_{\mathrm{B}'}$ through the crossing point v_{X} at the speed $\{s_1^i, s_2^i\}$ and $\{s_1^{i'}, s_2^{i'}\}$, respectively. The superscript and subscript of s are the sUAS index and the segment index, respectively)

section. Note that, in this study we assume that all sUASs are the same type to analyse urban airspace with less decision parameters.

5.3.2 Inner Loop: Single sUAS Flight Planning Problem

In this subsection, our focus is on flight planning a finite set of sUASs in a given route network-based urban airspace.

Definition 13 Let $\mathcal{A} = \{\alpha_1, \alpha_2, \ldots, \alpha_i\}$ be a finite set of sUASs. The set is partitioned

into two sets: a set of sUASs to be planned $\mathcal{A}^+$, and a set of planned sUASs $\mathcal{A}^-$ such that $\mathcal{A}^+ \cup \mathcal{A}^- = \mathcal{A}$, $\mathcal{A}^+ \cap \mathcal{A}^- = \emptyset$.

In the urban airspace, each sUAS flies from its origin to its destination. A feasible flight route for each sUAS in the urban airspace is defined as follows:

Definition 14 As given in the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$, sUAS $\alpha_i \in \mathcal{A}^+$ to be routed in $\mathcal{G}$, and the initial waypoint $v_1^i \in \mathcal{V}$ and the final waypoint $v_{n_i}^i \in \mathcal{V}$ construct a flight route denoted by $p(\alpha_i)$ in $\mathcal{G} = (\mathcal{E}, \mathcal{V})$, which is defined by a sequence of waypoints.

Thus, in the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$, there can be an abundance of flight route candidates denoted by $\mathcal{C}$ that satisfy the conditions as described in Definition 14. Through Definition 14, each flight route candidate $p(\alpha_i) \in \mathcal{C}$ can be given a corresponding flight route (a set of waypoints) $p(\alpha_i) : v_1^i, v_2^i, \ldots, v_{n_i}^i$ where n_i is the number of waypoints sUAS α_i traverses as shown in Figure 5.2. For each sUAS' flight route $p(\alpha_i)$, there is a set of segments connecting the waypoints, $\mathcal{E}(p(\alpha_i)) : e_{(1,2)}^i, e_{(2,3)}^i, \ldots, e_{(n_i-1,n_i)}^i$.

For each flight route candidate, it is assumed that a performance index for a set of segments $\mathcal{E}(p(i))$ can be quantified as a set of positive numeric weighting values, as follows:

$$\mathcal{W}(p(\alpha_i)) : w_1^i, w_2^i, \ldots, w_{n_i-1}^i.$$

Then, the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$ is transformed into a weighted directed graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W})$ by assigning a weight to each segment for all flight routes. In the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W})$, each flight route can be estimated by summing all weights of $\mathcal{W}(p(\alpha_i))$, as follows:

$$\mathcal{T}(p(\alpha_i)) = \sum_{j=1}^{n_i-1} w_j^i. \tag{5.1}$$

Based on the airspace graph, $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W})$, the routing problem that optimises a performance index can be defined as follows:

Definition 15 Given an airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W})$ and corresponding flight route candidates $\mathcal{C}$, the routing problem is defined as finding a flight route (or a sequence of waypoints) such that

$$p^*(\alpha_i) = \underset{p(\alpha_i) \in \mathcal{C}}{\operatorname{argmin}} \mathcal{T}(p(\alpha_i)). \tag{5.2}$$

The optimal flight route $p^*(\alpha_i)$ can be found by using the well-known shortest path algorithms such as Dijkstra's algorithm or A* algorithm or exhaustive search algorithm [40]. Although the optimal flight route $p^*(\alpha_i)$ can be obtained according to Definition 15, if the airspace users want to manage the separation between each sUAS in sequential approaches, the optimality of the obtained flight route might be disturbed in the following separation management stages. One possible option is to solve the multiple sUASs flight planning problem using numerical trajectory optimisation approaches, which is not scalable although the approaches provide high fidelity flight plans to each sUAS. Such approaches can either require additional processes for managing separation, which causes the optimality issues, or cause scalability issues.

Our objective of formulating the flight planning problem is to find a flight route and its speed profile that minimises the flight time of each sUAS while satisfying the separation requirement. The main idea for achieving the objective is to assign a weight to each segment (edge) of the airspace graph where the weight is expressed as a flight time. Then, each sUAS' speed profile also can be obtained by finding its flight route in the airspace graph as defined in Definition 15. We set a flight time of each sUAS as a weight $w \in \mathcal{W}$ to each segment $e \in \mathcal{E}$ of the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W})$. Note that, a flight time and speed of each sUAS are mutually interchangeable using geographic data included in the airspace graph. Where each sUAS' speed profile is determined within a feasible speed range of each sUAS. Another issue we have pursued is to satisfy the separation requirement between every pair of sUASs. To assign a flight time that satisfies the separation requirement to each edge of $\mathcal{G}$, we need the time data of the set of planned sUASs including arrival times of each sUAS at each waypoint as discussed in Section.5.3.1. The time data T will be included in the airspace graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W}, T)$, and be used to calculate weights of the airspace graph. By finding a solution of the airspace graph $\mathcal{G}$, then, we can obtain a flight route and speed profile of the sUAS while satisfying the separation requirement simultaneously. Each sUAS has different weights $\mathcal{W}_i$ because of different specifications such as sUAS' feasible speed range or the separation requirement as well as the origin and destination. Therefore, sUAS $\alpha_i \in \mathcal{A}^+$ has its unique airspace graph $\mathcal{G}_i$.

Calculating flight time weights $\mathcal{W}_{i,t}$ on $\mathcal{G}_i$ for separation-compliant speed profiles

We propose a weighting scheme to be applied to the airspace graphs $\mathcal{G}_i$ to fulfil our objectives. First of all, we introduce a set of constant speed for each sUAS $s(\alpha_i)$: $s_1^i, s_2^i, \ldots, s_{n_i-1}^i$ where each element of the set $s(\alpha_i)$ must be within a feasible speed range $[s_{\min}^i \; s_{\max}^i]$, $\forall \alpha_i \in \mathcal{A}$. Furthermore, we define a set of flight distances $d(\alpha_i)$: $d_1^i, d_2^i, \ldots, d_{n_i-1}^i$. using the geographical data allowing to compute the set $s(\alpha_i), \forall \alpha_i \in \mathcal{A}^+$.

Let $u_j^i = 1/s_j^i, \forall j \in \{1, 2, \ldots, n_i - 1\}, \forall \alpha_i \in \mathcal{A}^+, u_{\max}^i = 1/s_{\min}^i$, and $u_{\min}^i = 1/s_{\max}^i$, we formulate a linear programming problem and solve the problem to obtain flight time weight $\mathcal{W}_{i,t}$, $\forall \alpha_i \in \mathcal{A}^+$ as follows:

$$\min \quad \sum_{j=1}^{n_i-1} u_j^i d_j^i \tag{5.3}$$

$$\text{s.t.} \quad d_1^i u_1^i + d_2^i u_2^i + \cdots + d_{n_i-1}^i u_{n_i-1}^i \geq \max(T(v_{n_i}^i)) + t_{SP}^{(i',i)} \tag{5.4}$$

$$d_1^i u_1^i + d_2^i u_2^i + \cdots + d_{n_i-2}^i u_{n_i-2}^i \geq \max(T(v_{n_i-1}^i)) + t_{SP}^{(i',i)} \tag{5.5}$$

$$\vdots$$

$$d_1^i u_1^i + d_2^i u_2^i \geq \max(T(v_3^i)) + t_{SP}^{(i',i)} \tag{5.6}$$

$$d_1^i u_1^i \geq \max(T(v_2^i)) + t_{SP}^{(i',i)} \tag{5.7}$$

$$u_{\min}^i \leq u_1^i, u_2^i, \cdots, u_{n_i}^i \leq u_{\max}^i \tag{5.8}$$

where $t_{SP}^{(i',i)}$ is the minimum time-based separation requirement between preceding sUAS $\alpha_{i'}$ and following sUAS α_i at waypoints as shown in Figure 5.3 and Figure 5.4. The objective function Equation (5.3) is to minimise the sum of flight time of each segment. The constraints Equation (5.4) $\sim$ Equation (5.7) are for satisfying the minimum separation requirement at each waypoint. In Figure 5.2, for example, if $\mathcal{E}(p(\alpha_i))$

consists of $n_i - 1$ segments for sUAS α_i, each segment requires the flight time that satisfies the separation requirement. The left-hand side of each constraint of Equation (5.4) - Equation (5.7) is the flight time to the segment of each constraint, which must be greater than and equal to the time that satisfies the separation requirement. For the constraints, flight distances between waypoints and the time data are required. The time data is stored in T for each waypoint $v \in \mathcal{V}$, and T is updated every time a sUAS is allocated and shared with sUAS $\alpha_i \in \mathcal{A}^+$.

$$T(v) = \{t_1^v, t_2^v, \ldots, t_A^v\} \tag{5.9}$$

where A is the number of elements in $\mathcal{A}$. Then, the optimisation problem we formulated can be solved by any well-known linear programming algorithm. In this study, we utilise the interior-point algorithm. In Equation (5.8), decision variables, $s_1^i, s_2^i, \cdots, s_{n_i-1}^i$, are constant speeds for the flight segment, which are transformed into flight times, and assigned into weights $\mathcal{W}_{i,t} : w_{1,t}^i, w_{2,t}^i, \cdots, w_{n_i-1,t}^i$ for each flight route candidate of sUAS $\alpha_i \in \mathcal{A}^+$. Therefore, a solution to the airspace graph $\mathcal{G}_i = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}, T)$ can simultaneously provide a flight route and its speed profile for sUAS $\alpha_i, \forall \alpha_i \in \mathcal{A}^+$ while satisfying minimum separation at all times.

We also construct flight distance weights $\mathcal{W}_d$ on $\mathcal{G}_i$ to utilise the flight distance d as a second criterion. The weights $\mathcal{W}_d$ are the same for all sUASs, which is necessary to prioritise for the multiple sUASs' flight planning problem when two or more sUASs arrive at the same destination at the same time, more details about the priority are in Section.5.3.3. The single sUAS flight planning problem that motivated this study can be formulated as follows:

Problem 4 Given an unique airspace graph $\mathcal{G}_i = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}, \mathcal{W}_d, T)$, sUAS α_i in $\mathcal{G}_i$, and its origin $v_1^i \in \mathcal{V}$ and destination $v_{n_i}^i \in \mathcal{V}$ reachable from the origin, construct a flight route $p(\alpha_i)$ and a speed profile $s(\alpha_i), \forall \alpha_i \in \mathcal{A}^+$ such that

- the separation requirement is satisfied from the planned sUAS $\forall \alpha_i \in \mathcal{A}^-$,
- the speed profile of each sUAS must be within its feasible speed range,
- and the airspace graphs are updated every time when a sUAS $\alpha_i \in \mathcal{A}^+$ is planned.

In this study, we solve **Problem 4** using Dijkstra's algorithm [40].

5.3.3 Outer Loop: Multiple sUASs Flight Planning

In Section.5.3.2, our main idea is that the single sUAS flight planning problem can be formulated by assigning the separation satisfied flight time weights in the airspace graph. By doing so, the problem is able to not only manage separation between every pair of sUASs but also consider operational factors such as the feasible speed ranges, the separation requirements. However, the separation of each sUAS only satisfies the separation requirement from a set of allocated sUASs. In this Section, we address the multiple sUASs flight planning problem by solving each single sUAS flight planning problem in an iterative way. At each iteration, each sUAS finds the optimal flight route including its speed profile by solving Problem 4 (inner loop), and one of the sUASs is allocated using a criterion-based algorithm (outer loop). In this study, we utilise the First Come First Served (FCFS) algorithm, and the Last Come First Served (LCFS) algorithm for the outer loop algorithm.

Once inputs of the multiple sUASs flight planning problem are given, Algorithm 4 starts the flight planning process until $\mathcal{A}^+$ is empty (Line 3). The algorithm first generates airspace graphs $\mathcal{G}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_d, T^k), \forall \alpha_i \in \mathcal{A}^+$ (Line 2). In order to reduce the complexity of the problem caused by high degree of the airspace graphs, we select the h-shortest routes for $\mathcal{G}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_d, T^k), \forall \alpha_i \in \mathcal{A}^+$ by using Yen's algorithm, and generate $\bar{\mathcal{G}}_i^k$ only considering the h-shortest routes (Line 4-5) [41]. Then, each sUAS has its unique airspace graph that contains flight time weights $\mathcal{W}_{i,t}^k$ determining the optimal flight route with its speed profile. The optimal flight route $p^*(\alpha_i)$ and its speed profile $s(\alpha_i)$ are obtained by using Dijkstra's algorithm, $\forall \alpha_i \in \mathcal{A}^+$ (Line 6) [42]. sUAS α_i^* that arrives first to its destination is obtained by using the FCFS algorithm (Line 7), and allocated (Line 15). The flight plan of sUAS α_i^* is transferred from $\mathcal{A}^+$ to $\mathcal{A}^-$ with its route and speed profile (Line 12). Note that, in this study, α_i refers to a flight as well as a sUAS. In case of more than two sUASs are arriving to the same destination at the same time (Line 8), the flight distance weights $\mathcal{W}_d$ are utilised to allocate sUAS $\alpha_i^\dagger$ amongst them (Line 9-13). In either case, a sUAS is allocated and

its time data is shared with sUAS α_i, $\forall \alpha_i \in \mathcal{A}^+$. Based on the time data each airspace graph $\mathcal{G}_i^k$ is updated, $\forall \alpha_i \in \mathcal{A}^+$ (Line 20). Note that, to change the outer loop algorithm to other criterion-based algorithms such as the LCFS algorithm, users are only required to change the outer loop algorithm Line 7 and Line 10 to what the users want.

We illustrate an example in Figure 5.5 for a better understanding of the outer loop concept of the proposed algorithm with the FCFS algorithm as the outer loop algorithm. In the first iteration, each sUAS generates its airspace graph $\mathcal{G}_i^1$ and $\bar{\mathcal{G}}_i^1$ is found, $\forall \alpha_i \in \mathcal{A}^+$. Through the inner loop process, each sUAS's $p^*(\alpha_i)$ and $s^*(\alpha_i)$ is determined. The first come sUAS α_2 is planned using the FCFS algorithm. In the second iteration, sUAS α_A is planned through the same process as the first iteration. In the $(A)^{th}$ iteration, sUAS α_1 is planned finally. Although in the example above we assume that the first come sUAS has the priority to be planned, the criteria or the outer loop algorithm to determine the priority can be changed.

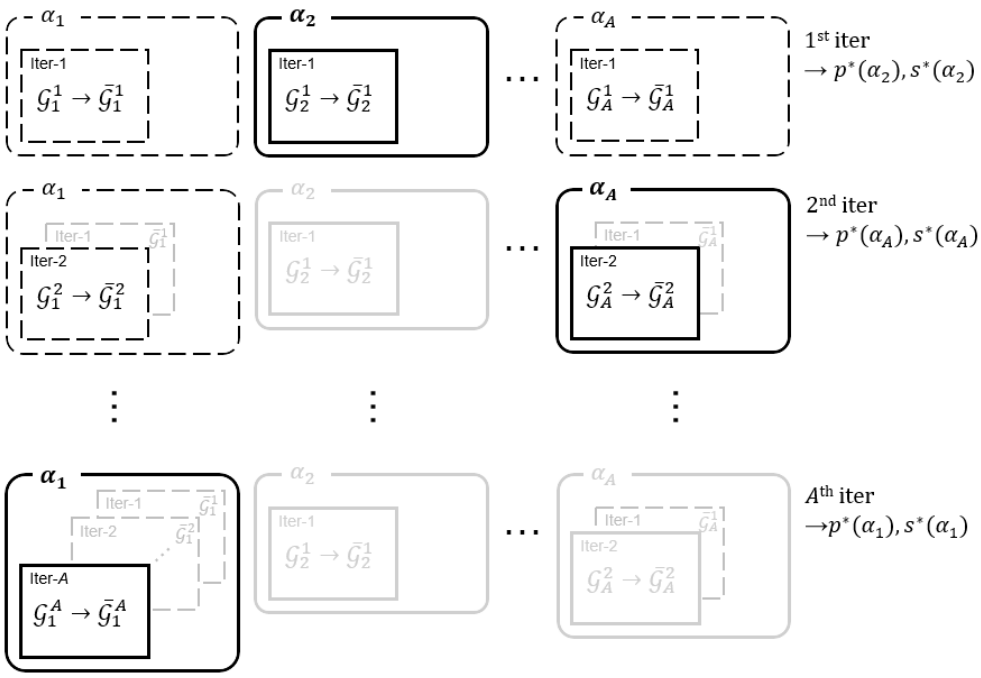


Figure 5.5: Iteratively generated airspace graph concept for multiple sUASs in Algorithm 4

5.3.4 Flight Planning Example for 1-to-$\mathcal{M}$ Case (Last-mile Delivery with a Single Retail Point)

This example is a sample of the Monte Carlo simulations for 1-to-$\mathcal{M}$ case, and its simulation settings are described at the beginning of Section 5.4. Put briefly, thirty sUASs traverse from the retail point (E68) to each predetermined sUAS' service point (E1$\sim$E67) in Figure 5.7. We apply both the FCFS algorithm and the LCFS algorithm to this example, and thirty flight plans for each case are shown in Table 5.1 and Table 5.2, respectively. The results show each sUAS' separation requirement satisfied flight route and its speed profile for each outer loop algorithm. Due to 5 seconds departure separation, 5 seconds horizontal separation and the single retail point of this case, all sUASs find their flight routes with the maximum speed without any conflicts for both cases. For the same reasons, most of the flight routes are same as the shortest path of each sUAS. Table 5.3 shows comparison results between the FCFS algorithm and the LCFS algorithm for this example. The total flight time and the total flight distance of thirty sUASs for the FCFS algorithm case is almost same as the LCFS algorithm case. However, the mission completion time for the LCFS algorithm case is 34.5% more efficient than the FCFS algorithm case. Therefore, although the total flight time and the total flight distance are very similar for both cases, the LCFS algorithm case is more efficient with regard to the urban airspace operating time. Both airspace users and airspace service providers prefer to operate as many sUASs as possible in a given time within a given urban airspace. However, if there are more than two urban airspace users with more than two retail points, the urban airspace usage priority will be an essential factor to be considered for the flight planning problems, which is a similar situation as today's airspace for commercial aircraft.

Advantages of the proposed algorithm are as follows: able to adopt different sequencing algorithms; providing separation-compliant speed profiles that satisfy each sUAS's performance; applicable to various types of route networks; enlarge a solution searching space by finding a flight route and speed profile of each sUAS simultaneously; and fast computational time. Therefore, we expect that the advantages of Algorithm 4, which provides not only the flight route but also the detailed schedule, will be

appropriate to analyse the capacity of the high-density VLL urban airspace on which many sUASs with short flight times, and many service providers.

5.3.5 Four Metrics for the Capacity Estimation

In the ATM system, typically, the airspace capacity is defined as the maximum number of aircraft that can be accommodated in a given airspace at any point of time, while throughput is defined as the number of aircraft that land to an airport over a specific time window. In sUASs flight in high-density VLL urban airspace, however, it is expected that it will be difficult to construct and to negotiate multiple sUASs' flight plans a few days in advance like the flight plans in the ATM system and it will be complicated to conduct separation management through the air traffic controllers. Compared to commercial aircraft, also, it is expected that very short flight times, many landing points, many flights, and fewer operators. Thus, we can expect that one of options to analyse the urban airspace capacity based on the detailed flight plans with regard to the flight time to complete the given mission, the total flight time, the total flight distance, and the number of conflicts. We utilise such detailed flight plans to define four metrics to analyse the urban airspace capacity.

Total flight time (M_{tft})

Intuitively, one metric to analyse the urban airspace capacity is the total flight time for given missions. The total flight time can be calculated as follows:

$$M_{tft} = \sum_{i=1}^{A} \sum_{j=1}^{n_i - 1} w_{j,t}^{i} \tag{5.10}$$

where A is the number of sUASs to be planned. M_{tft}, also, can be utilised as a metric to measure operating costs for the sUAS operators.

Mission completion time (M_{mct})

Once missions are given to a set of sUASs, each sUAS generates its separation-compliant flight plan. Then, the mission completion time is the latest arrival time at a service point

or a retail point of the last sUAS, which can be found as follows:

$$M_{mct} = \underset{\forall v \in \mathcal{V}}{\operatorname{argmax}} \, T^A(v) \tag{5.11}$$

The meaning of this metric can be used as an indicator to determine which algorithms are effective to operate the route network-based urban airspace under the same conditions. This will be one of the most important metrics for the urban airspace operators.

Total flight distance (M_{tfd})

The flight distance is a commonly used as a criterion for analysing the flight performance to find the optimal path in the urban airspace. Each sUAS' flight distance, which is used for the second criterion in Algorithm 4, can be obtained from the geometry information. The total flight distance can be calculated as follows:

$$M_{tfd} = \sum_{i=1}^{A} \sum_{j=1}^{n_i - 1} d_j^i \tag{5.12}$$

Normalised number of conflicts (M_{cnf})

Algorithm 4 generates a flight plan that only fulfils the destination separation requirement within the permissible speed range when the algorithm could not find a flight route that satisfies the separation requirement during the whole flight as well as the destination separation requirement and the permissible speed range. Namely, the sUAS could conflict with other sUASs during the flight based on the obtained flight plans. Although this study does not provide an overtaking manoeuvres, it is assumed that overtaking manoeuvres via vertical manoeuvring is sufficiently possible using current CD&R (Conflict Detection and Resolution) technologies. Then, the number of conflicts is counted whenever a sUAS overtakes a preceding sUAS. Naturally, the number of conflicts increases with the number of sUASs in the urban airspace. We divide the number of conflicts by the total number of possible conflicts to normalise, and use it as a metric:

$$M_{cnf} = \frac{\textit{Number of conflicts}}{\textit{Total number of possible conflicts}} \tag{5.13}$$

where *Total number of possible conflicts* is $\frac{A \times (A+1)}{2}$.

Algorithm 4: Multiple sUASs flight planning algorithm

Input: $v_1^i, v_{n_i}^i \in \mathcal{V}, \forall \alpha_i \in \mathcal{A}$, airspace information

Output: $p^*(\alpha_i), s^*(\alpha_i), \forall \alpha_i \in \mathcal{A}$

$k = 1$

generate $\mathcal{G}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_d, T^k), \forall \alpha_i \in \mathcal{A}^+$

while $\mathcal{A}^+ \neq \emptyset$ **do**

  find h-shortest routes for $\mathcal{G}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_d, T^k), \forall \alpha_i \in \mathcal{A}^+$ (using Yen's algorithm);

  generate $\bar{\mathcal{G}}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}^k, \mathcal{W}_d, T^k)$ that only considers the best h routes, $\forall \alpha_i \in \mathcal{A}^+$;

  find $p^*(\alpha_i)$ of $\bar{\mathcal{G}}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}^k, T^k), \forall \alpha_i \in \mathcal{A}^+$ (using Dijkstra's algorithm);

  $\alpha_i^* \leftarrow \underset{\forall \alpha_i \in \mathcal{A}^+}{\operatorname{argmin}} p^*(\alpha_i)$ (using FCFS algorithm);

  if *There are more than two* α_i^* *exist* **then**

    find $p^*(\alpha_i)$ of $\bar{\mathcal{G}}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_d, T^k)$ amongst them (using Dijkstra's algorithm);

    $\alpha_i^\dagger \leftarrow \underset{\forall \alpha_i^*}{\operatorname{argmin}} p^*(\alpha_i)$ (using FCFS algorithm);

    allocate $\alpha_i^\dagger$;

    transfer $\alpha_i^\dagger$ from $\mathcal{A}^+$ to $\mathcal{A}^-$;

    share $\mathcal{A}^-$ with $\forall \alpha_i \in \mathcal{A}^+$;

  else

    allocate α_i^*;

    transfer α_i^* from $\mathcal{A}^+$ to $\mathcal{A}^-$;

    share $\mathcal{A}^-$ with $\forall \alpha_i \in \mathcal{A}^+$;

  end

  $k = k + 1$;

  update $\mathcal{G}_i^k = (\mathcal{E}, \mathcal{V}, \mathcal{W}_{i,t}^k, \mathcal{W}_d, T^{\mathrm{k}}), \forall \alpha_i \in \mathcal{A}^+$;

end

Table 5.1: Flight planning results for 30 flights for 1-to-$\mathcal{M}$ case (Outer loop: FCFS algorithm)

Flight ID	**Waypoint sequence (arrival time at each waypoint [sec]) speed profile [km/h]**
⋮	⋮
α_4	E68(180) - E24(197) - E17(217) - E1(244) - E2(268) - E3(280) - E4(293) 25 - 25 - 25 - 25 - 25 - 25
α_5	E68(220) - E24(237) - E17(257) - E1(284) - E2(308) - E3(320) - E4(333) - E5(345) 25 - 25 - 25 - 25 - 25 - 25 - 25
⋮	⋮
α_{22}	E68(130) - E32(147) - E33(169) - E34(181) - E35(193) - E36(206) 25 - 25 - 25 - 25 - 25
⋮	⋮

Table 5.2: Flight planning results for 30 flights for 1-to-$\mathcal{M}$ case (Outer loop: LCFS algorithm)

Flights ID	**Waypoint sequence (arrival time at each waypoint [sec]) speed profile [km/h]**
⋮	⋮
α_4	E68(110) - E24(127) - E17(147) - E1(173) - E2(197) - E3(210) - E4(223) 25 - 25 - 25 - 25 - 25 - 25
α_5	E68(70) - E24(87) - E17(107) - E1(133) - E2(157) - E3(170) - E4(183) - E5(195) 25 - 25 - 25 - 25 - 25 - 25 - 25
⋮	⋮
α_{22}	E68(160) - E32(177) - E33(198) - E34(211) - E35(223) - E36(236) 25 - 25 - 25 - 25 - 25
⋮	⋮

Table 5.3: Comparison between the FCFS algorithm and the LCFS algorithm for 1-to-$\mathcal{M}$ case

	FCFS algorithm	**LCFS algorithm**
Total flight time [sec]	7035.1	7031.1
Total flight distance [km]	19.3	19.3
Mission completion time [sec]	468.7	307.0
Number of conflict	0	0

5.4 Numerical Simulations

In this section, we conduct Monte Carlo (MC) simulations with two different outer loop algorithms (the FCFS algorithm, the LCFS algorithm) that are based on the proposed flight planning algorithm for the four drone delivery operation types. We investigate how the different operation types' performance and safety behave with regard to the different sequencing algorithms and the number of sUASs in the high-density VLL urban airspace using the suggested four metrics. It aims to provide an approach for choosing between resolution strategies for the urban airspace management to regulators, policymakers, and urban airspace designers.

Figure 5.6: Oldbrook, Milton Keynes, United Kingdom (*Google Earth Pro, 2018*)

The representative operation being considered in this study is drone delivery, especially the *last-mile delivery* and the *first-mile delivery*. *Last-mile delivery* is defined as the movement of items from transportation hubs (which are retailers in this study) to the final delivery destinations in which the destination are typically personal residences. *First-mile delivery* refers to the movement of goods from sellers to courier services who will take these goods to their final users.

A scenario for the simulations is based on the town 'Oldbrook', which is one of

the well-planned towns in Milton Keynes in the United Kingdom. Two-dimensional infrastructure data of the town is obtained from Google Earth Pro. The scenario has an area size of 0.98 km^2 as shown in Figure 5.6. Similar to the en-route airspace for commercial aircraft, in this route network, we construct two layers of nodes that are set above the roads at the height of 15 meters and 25 meters as shown in Figure 5.7 and Figure 5.8 for eastbound and westbound, respectively. We assume that each layer consists of 70 points (3 retail points (red pins) and 67 service points (green pins)) and 107 directed routes as shown in Figure 5.7 and Figure 5.8, and all sUASs are of the same type as shown in Table 5.4. The sUAS type can be changed and varied based on the missions and operators later. For the separation between sUAS, it is assumed that the time-based minimum vertical and horizontal separation requirement during the flight is 5 seconds. No hovering manoeuvre is allowed for any sUASs.

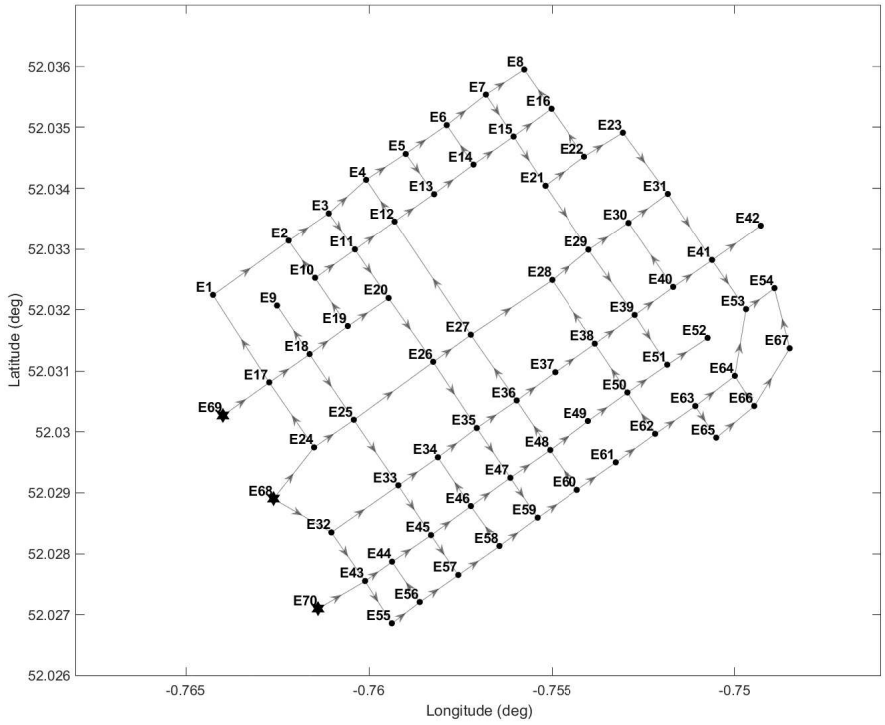


Figure 5.7: Eastbound at 15 meters, 3 retail points (E68, E69, E70) and 67 service points (E1∼E67) and 107 directed routes, (Oldbrook, Milton Keynes, United Kingdom)

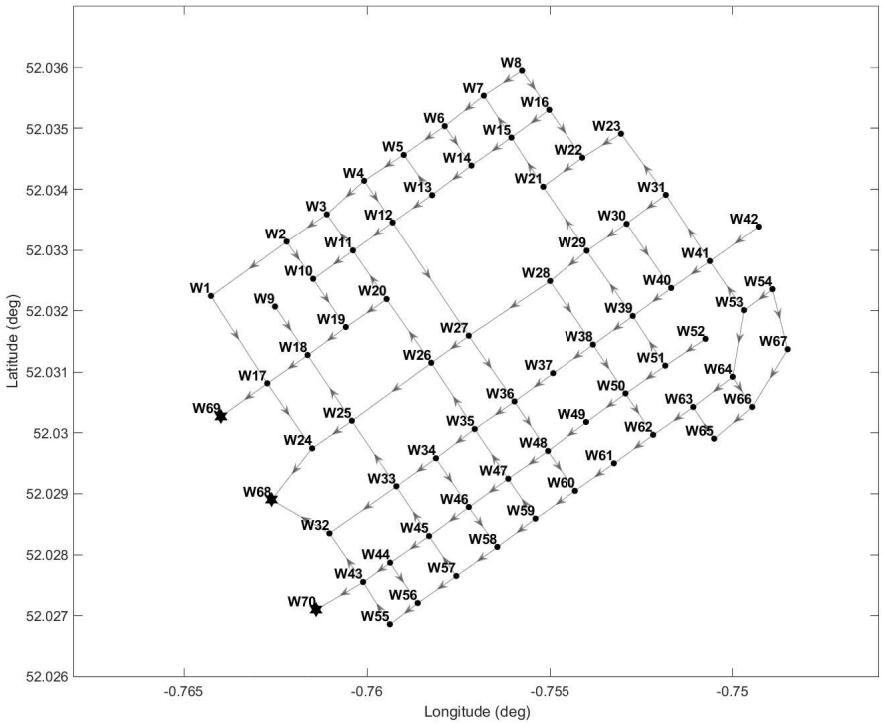


Figure 5.8: Westbound at 20 meters, 3 retail points (W68, W69, W70) and 67 service points (W1∼W67) and 107 directed routes, (Oldbrook, Milton Keynes, United Kingdom)

We conduct 100 MC simulations for each outer loop algorithms (the FCFS algorithm and the LCFS algorithm) by increasing the number of sUASs from 5 to 50 with 5 interval for the four drone delivery operation types (1-to-$\mathcal{M}$, $\mathcal{M}$-to-1, $\mathcal{N}$-to-$\mathcal{M}$, $\mathcal{M}$-to-$\mathcal{N}$) as shown in Table 5.5. At each simulation a service point and a retail point for each sUAS are randomly chosen except for the 1-to-$\mathcal{M}$ and $\mathcal{M}$-to-1 cases in which there is the single retail point E68 and W68, respectively. Once the route network, and each sUAS' retail point and service point are given which are the input of Algorithm 4, the algorithm generates each sUAS' flight plan as described in Section 5.3. From the output of the algorithm, we obtain M_{tft} (total flight time), M_{mct} (mission completion time), M_{tfd} (total flight distance), and M_{cnf} (normalised number of conflicts).

Both the FCFS algorithm results and the LCFS algorithm results quantify the route network-based urban airspace using the four metrics. For each operation type, we

Table 5.4: sUAS specification

	small UAS
Speed Range [km/h]	[5 25]
Endurance [sec]	900
Separation [sec]	5
Max Weight [kg]	25 (FAA Part 107)

Table 5.5: Eight Monte Carlo simulation cases

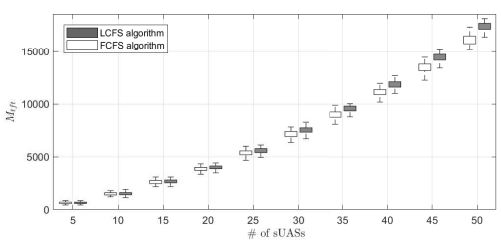
	FCFS algorithm	**LCFS algorithm**
1-to-$\mathcal{M}$ case	✓	✓
$\mathcal{M}$-to-1 case	✓	✓
$\mathcal{N}$-to-$\mathcal{M}$ case	✓	✓
$\mathcal{M}$-to-$\mathcal{N}$ case	✓	✓

analyse both results, and suggest one of the two outer loop algorithms that is more appropriate for the operation type.

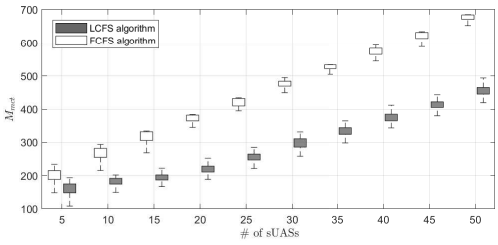
5.4.1 Case Study: 1-to-$\mathcal{M}$ Last-mile Delivery (Eastbound)

In the 1-to-$\mathcal{M}$ operation type, each sUAS traverses with an item from the single retail point (E68) to each sUAS' predetermined service point between E1$\sim$E67 in Figure 5.7 where we ignore two retail points (E69, E70) and two routes (E69$\rightarrow$E17, E70$\rightarrow$E43).

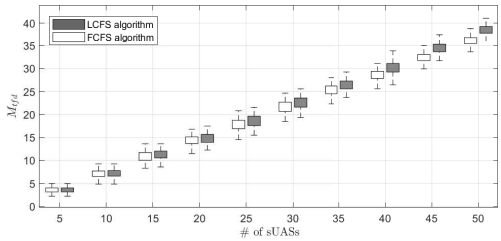
In both of the outer loop algorithms, multiple sUASs traverse at maximum speed due to the five second time interval between each sUAS departures at the single retail point (E68). Thus, there is no significant difference between the two algorithms in M_{tft} and M_{tfd} as shown in Figure 5.9a and Figure 5.9c, respectively. Also, the interval allows them to satisfy the separation requirement at all time, so that there is no conflict between every pair of the sUASs. However, Figure 5.9b shows that the LCFS algorithm



(a) M_{tft} (total flight time [sec])



(b) M_{mct} (mission completion time [sec])



(c) M_{mct} (total flight distance [km])

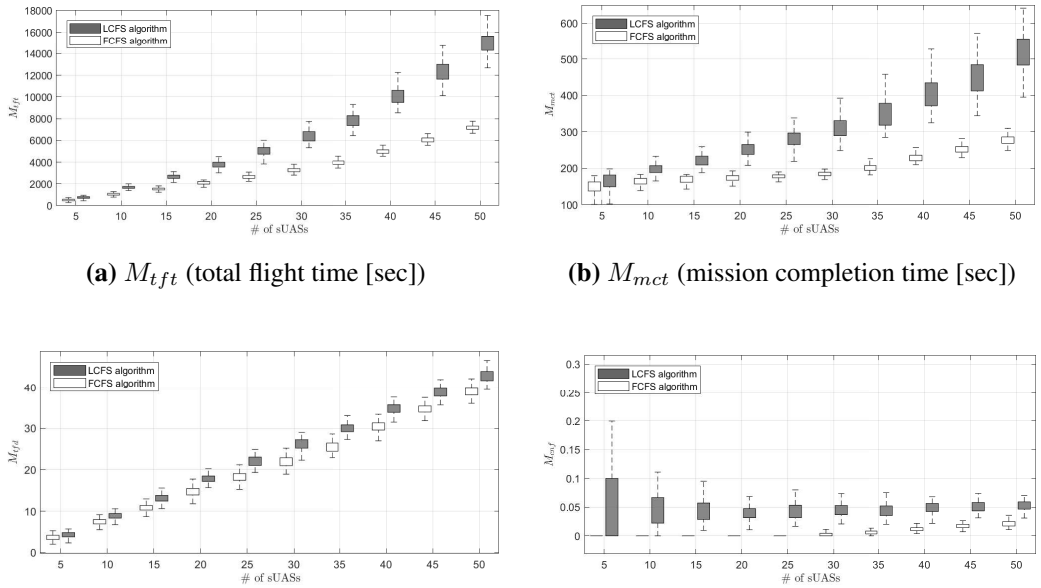
Figure 5.9: 1-to-$\mathcal{M}$ Case: Standard deviation comparisons between the FCFS algorithm and the LCFS algorithm for M_{tft}, M_{mct}, and M_{tfd}

results in more efficient mission completion times compared to the FCFS algorithm, which means the LCFS algorithm allows more sUASs to fly at a given time within the urban airspace. Therefore, in the operation type of 1-to-$\mathcal{M}$, the LCFS algorithm is efficient.

5.4.2 Case Study: $\mathcal{M}$-to-1 First-mile Delivery (Westbound)

In the $\mathcal{M}$-to-1 operation type, multiple sUASs fly from each sUAS' predetermined service point between W1$\sim$W67 to the single retail point (W68) in Figure 5.8 where we ignore two retail points (W69, W70) and two routes (W17$\rightarrow$W69, W43$\rightarrow$W70). We assume that all sUASs depart from each sUAS' service points at the same time to consider the worst case scenario.

Figure 5.10c shows that M_{tfd} of the LCFS algorithm is slightly higher than the FCFS algorithm, and Figure 5.10a shows that M_{tft} is rapidly raised as the number of sUASs increases compared to M_{tfd}. As a result, it is seen that the LCFS algorithm causes not only detours but also low flight speeds. These results are inextricably linked



(a) M_{tft} (total flight time [sec])

(b) M_{mct} (mission completion time [sec])

(c) M_{tfd} (total flight distance [km])

(d) M_{cnf}

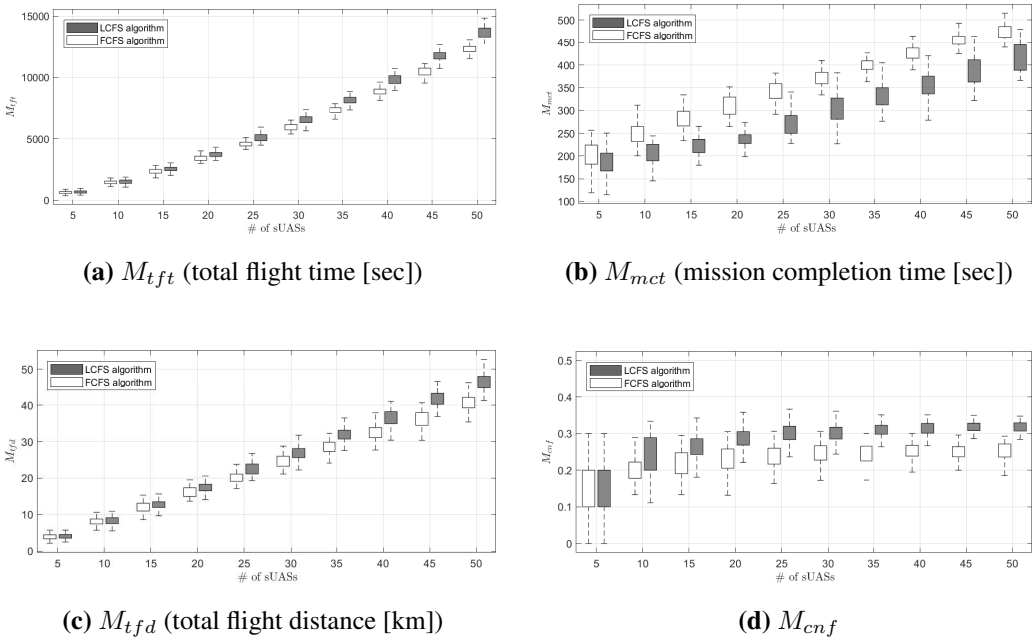
Figure 5.10: $\mathcal{M}$-to-1 Case: Standard deviation comparisons between the FCFS algorithm and the LCFS algorithm for $M_{tft}, M_{mct}, M_{tfd}$, and M_{cnf}

to M_{cnf} shown in Figure 5.10d, resulting in inefficient and unpredictable results for M_{cnf}. In all respects, therefore, the FCFS algorithm is better than the LCFS algorithm for $\mathcal{M}$-to-1 operation type for the urban airspace.

5.4.3 Case Study: $\mathcal{N}$-to-$\mathcal{M}$ Last-mile Delivery (Eastbound)

In the $\mathcal{N}$-to-$\mathcal{M}$ operation type, each sUAS traverses with an item from the predetermined retail points (E68, E69, E70) to the each sUAS' predetermined service point from E1$\sim$E67 in Figure 5.7. It is assumed that each retail point has five seconds minimum departure interval.

From the safety and flight performance perspective, the FCFS algorithm results in more efficient results for M_{tft}, M_{tfd}, and M_{cnf} results as shown in Figure 5.11. However, M_{mct} shows that the LCFS algorithm can allow to accommodate more sUASs in the given space within a given time. Therefore, in the case of $\mathcal{N}$-to-$\mathcal{M}$ for the urban airspace, the algorithm is dependent on the situation as shown in Figure 5.11. For



(a) M_{tft} (total flight time [sec])

(b) M_{mct} (mission completion time [sec])

(c) M_{tfd} (total flight distance [km])

(d) M_{cnf}

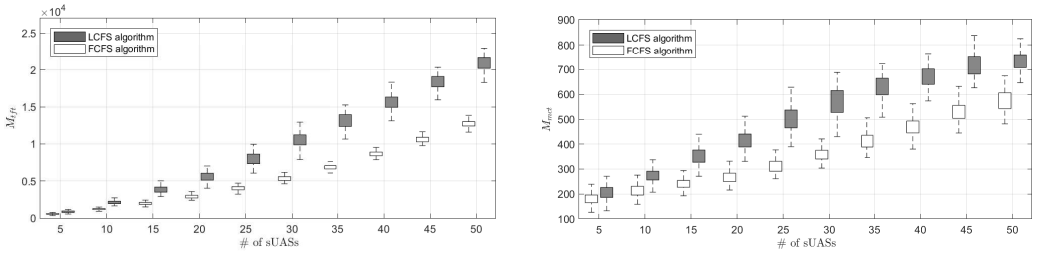
Figure 5.11: $\mathcal{N}$-to-$\mathcal{M}$ Case: Standard deviation comparisons between the FCFS algorithm and the LCFS algorithm for $M_{tft}, M_{mct}, M_{tfd}$, and M_{cnf}

example, it is appropriate to use the LCFS algorithm when a large number of items must be delivered in the given urban airspace, while the FCFS algorithm will be used for safety and individual performance when there is more room in the given urban airspace.

5.4.4 Case Study: $\mathcal{M}$-to-$\mathcal{N}$ First-mile Delivery (Westbound)

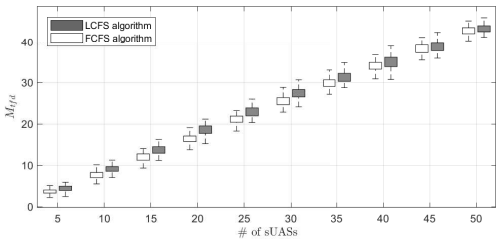
In the $\mathcal{M}$-to-$\mathcal{N}$ operation type, each sUAS traverses with an item from each sUAS' pre-determined service point between W1$\sim$W67 to the predetermined retail points (W68, W69, W70) in Figure 5.8. We assume that all sUASs depart from each sUAS' service points at the same time to consider the worst case scenario.

Figure 5.12 shows that the results of M_{tft}, M_{tfd}, and M_{mct} are very similar to the results of the $\mathcal{M}$-to-1 operation type. Namely, the FCFS algorithm is appropriate for this case with regard to the three metrics. However, the M_{cnf} result of the LCFS algorithm approaches to 0.05, while in the FCFS algorithm results steadily increases.

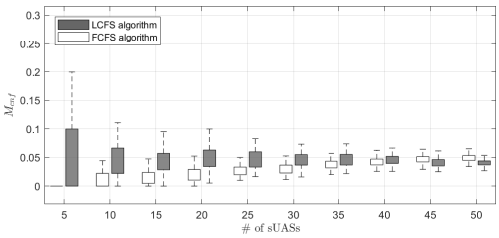


(a) M_{tft} (total flight time [sec])

(b) M_{mct} (mission completion time [sec])



(c) M_{tfd} (total flight distance [km])



(d) M_{cnf}

Figure 5.12: $\mathcal{M}$-to-$\mathcal{N}$ Case: Standard deviation comparisons between the FCFS algorithm and the LCFS algorithm for $M_{tft}, M_{mct}, M_{tfd}$, and M_{cnf}

This is because of the overtaking manoeuvres caused by the outer loop algorithm. The FCFS algorithm finds flight plans while satisfying the minimum separation requirement when there are a small number of sUASs. However, as the number of sUASs increases, M_{cnf} is increasing because of a bottleneck phenomenon around the retail points. In the case of the LCFS algorithm, even when a small number of sUASs are flying, the M_{cnf} always shows a similar tendency because the following sUASs may outstrip the preceding sUASs. Because of the same reason, when there are a small number of sUASs in the urban airspace, the variation is large as shown in Figure 5.12d. Due to the tendency, the result of the FCFS algorithm shows a higher M_{cnf} than the LCFS algorithm's one from when the number of sUASs is 45. From this result, we suggest applying either the FCFS algorithm or LCFS algorithm to the airspace users according to the number of sUASs in the urban airspace.

5.4.5 Comparison in M_{mct} and M_{cnf} Between the FCFS Algorithm and the LCFS Algorithm

In order to complement Figure 5.9 $\sim$ Figure 5.12, we provide Table 5.6 to show the comparison in M_{mct} and M_{cnf} between the FCFS algorithm and the LCFS algorithm. The metric, M_{cnf}, is the cause of increasing M_{tft}, and M_{tfd}. Namely, M_{tft} and M_{tfd}, which also indicate sUASs' performance, are directly related to M_{cnf}. Mission completion time, M_{mct}, is an indicator of the performance of the urban airspace within a given period of time, although its relationship with M_{tft} and M_{tfd} is not prominent. From the results, the following analysis can be made: 1) For the four operation types, the FCFS algorithm generates more efficient results than the LCFS algorithm for M_{tft} and M_{tfd}. This is because sUASs with short flight times complete the flight first, and then the other sUASs can fly in relaxed urban airspace at high speed; 2) For the 1-to-$\mathcal{M}$ operation type, the LCFS algorithm shows very efficient results for M_{mct}. This indicates that the LCFS algorithm efficiently uses the urban airspace for a given time period; and 3) From the standard deviation of the results except for the 1-to-$\mathcal{M}$ operation type, the FCFS algorithm is easier to predict each metric as the number of sUASs increases than the LCFS algorithm. The main reason why the standard deviation of the LCFS algorithm results are high is that the variation of each sUAS' speed profile dependent on the randomly selected starting point and destination point is very large.

Table 5.6: Comparison in M_{mct} and M_{cnf} between the FCFS algorithm and the LCFS algorithm [%]

		# of sUASs	**5**	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**	**50**
1-to-$\mathcal{M}$	FCFS	M_{mct}	200.7	263.8	318.6	369.6	421.2	474.6	523.8	573.1	621.4	674.8
		M_{cnf}	0	0	0	0	0	0	0	0	0	0
	LCFS	M_{mct}	162.0	179.5	194.9	220.5	254.2	296.2	333.7	374.2	413.9	457.4
		M_{cnf}	0	0	0	0	0	0	0	0	0	0
$\mathcal{M}$-to-1	FCFS	M_{mct}	147.9	164.6	168.0	173.8	177.7	185.0	201.8	229.2	251.9	278.9
		M_{cnf}	0.0000	0.0000	0.0001	0.0004	0.0012	0.0033	0.0061	0.0122	0.0169	0.0213
	LCFS	M_{mct}	162.9	199.4	222.7	252.5	283.4	312.4	349.9	404.3	452.6	518.6
		M_{cnf}	0.0530	0.0453	0.0454	0.0383	0.0444	0.0456	0.0441	0.0485	0.0507	0.0525
$\mathcal{N}$-to-$\mathcal{M}$	FCFS	M_{mct}	200.7	247.4	282.6	311.5	341.6	372.0	398.6	424.8	455.5	474.7
		M_{cnf}	0.1350	0.1991	0.2176	0.2303	0.2373	0.2442	0.2433	0.2521	0.2493	0.2502
	LCFS	M_{mct}	188.5	205.6	221.8	239.6	273.4	302.9	331.8	354.9	387.4	416.0
		M_{cnf}	0.1840	0.2402	0.2639	0.2827	0.2998	0.3011	0.3101	0.3136	0.3175	0.3186
$\mathcal{M}$-to-$\mathcal{N}$	FCFS	M_{mct}	178.6	212.9	242.7	269.7	313.4	358.4	413.0	469.4	530.1	572.6
		M_{cnf}	0.0140	0.0138	0.0157	0.0212	0.0271	0.0310	0.0373	0.0428	0.0471	0.0495
	LCFS	M_{mct}	206.5	273.6	355.4	415.8	498.1	569.4	627.5	671.9	715.1	735.4
		M_{cnf}	0.0390	0.0389	0.0442	0.0490	0.0463	0.0468	0.0458	0.0461	0.0416	0.0405

5.5 Conclusions

This work investigated the capacity analysis for the route network-based urban airspace with regard to the four drone delivery operation types. Two different flight planning algorithms were compared for each operation type using the MC simulations. From the analysis, the following implications can be made. First, even on the same operation type, it is required that the flight planning algorithm can consider operation factors such as the number of retail points, number of service points, number of sUASs, etc., and that fast-time simulation using the flight planning algorithm in order to analyse the route network-based urban airspace capacity for the efficient operations. Secondly, in urban airspace capacity estimation, where safety and performance analysis are primarily considered, it is needed to analyse the capacity with regard to airspace users' priorities. Finally, such analysis and results could be used for designing structures depending on urban airspace situations and environments.

We expect that the results could give the following support to the policymakers, urban airspace designers, and regulators: 1) When the stakeholders configure a new structure of the urban airspace, they can utilise the metrics to estimate the capacity according to the situation and environment; 2) It is easy to analyse the entry of new stakeholders by increasing the number of retail points; and 3) It is available as a test bed for new sequence allocation algorithms.

References

[1] N. Lavars, "Amazon to begin testing new delivery drones in the US," *New Atlas*, 2015.

[2] Amazon.com Inc., "Determining Safe Access with a Best- Equipped, Best-Served Model for Small Unmanned Aircraft Systems," *NASA UTM 2015: The Next Era of Aviation*, 2015.

[3] D. Simmons, "Rwanda begins Zipline commercial drone deliveries," *BBC World News*, 2016.

[4] C. A. Thiels, J. M. Aho, S. P. Zietlow, and D. H. Jenkins, "Use of unmanned aerial vehicles for medical product transport," *Air Medical Journal*, vol. 34, no. 2, pp. 104–108, 2015.

[5] L. A. Haidari, S. T. Brown, M. Ferguson, E. Bancroft, M. Spiker, A. Wilcox, R. Ambikapathi, V. Sampath, D. L. Connor, and B. Y. Lee, "The economic and operational value of using drones to transport vaccines," *Vaccine*, vol. 34, no. 34, pp. 4062–4067, jul 2016.

[6] J. Hoekstra, S. Kern, O. Schneider, F. Knabe, and B. Lamiscarre, "Metropolis – Urban Airspace Design," Technical University of Delft National, Tech. Rep., 2015.

[7] P. Kopardekar, "Unmanned Aerial System (UAS) Traffic Management (UTM): Enabling Low-Altitude Airspace and UAS Operations," National Aeronautics and Space Administration, Tech. Rep. April, 2014.

[8] T. Prevot, J. Rios, P. Kopardekar, J. E. Robinson III, M. Johnson, and J. Jung, "UAS Traffic Management (UTM) Concept of Operations to Safely Enable Low Altitude Flight Operations," *16th AIAA Aviation Technology, Integration, and Operations Conference*, no. June, pp. 1–16, 2016.

[9] V. Bulusu, R. Sengupta, V. Polishchuk, and L. Sedov, "Cooperative and non-cooperative UAS traffic volumes," *2017 International Conference on Unmanned Aircraft Systems, ICUAS 2017*, pp. 1673–1681, 2017.

[10] L. Sedov, V. Polishchuk, and V. Bulusu, "Sampling-based capacity estimation for unmanned traffic management," *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, vol. 2017-Septe, 2017.

[11] M. F. B. Mohamed Salleh, C. Wanchao, Z. Wang, S. Huang, D. Y. Tan, T. Huang, and K. H. Low, "Preliminary Concept of Adaptive Urban Airspace Management for Unmanned Aircraft Operations," in *2018 AIAA Information Systems-AIAA Infotech @ Aerospace*, no. January. Reston, Virginia: American Institute of Aeronautics and Astronautics, jan 2018, pp. 1–12.

[12] I. V. Laudeman, S. G. Shelden, R. Branstrom, and C. L. Brasil, "Dynamic Density: An Air Traffic Management Metric," National Aeronautics and Space Administration, Tech. Rep. April, 1998.

[13] A. Majumdar, W. Ochieng, and J. Polak, "Estimation of European Airspace Capacity from a Model of Controller Workload," *Journal of Navigation*, vol. 55, no. 03, sep 2002.

[14] A. Majumdar, W. Y. Ochieng, J. Bentham, and M. Richards, "En-route sector capacity estimation methodologies: An international survey," *Journal of Air Transport Management*, vol. 11, no. 6, pp. 375–387, nov 2005.

[15] A. Klein, L. Cook, B. Wood, and D. Simenauer, "Airspace capacity estimation using flows and Weather-Impacted Traffic Index," in *2008 Integrated Communications, Navigation and Surveillance Conference*. IEEE, may 2008, pp. 1–12.

[16] A. J. H. Qinetiq, E. Care, and I. Metrics, "Assessing the Capacity of Novel ATM Systems," in *4th USA/Europe Air Traffic Management R&D Seminar*, no. December, 2001.

[17] V. Bulusu, V. Polishchuk, R. Sengupta, and L. Sedov, "Capacity Estimation for Low Altitude Airspace," *17th AIAA Aviation Technology, Integration, and Operations Conference*, no. June, pp. 1–15, 2017.

[18] L. Sedov and V. Polishchuk, "Centralized and Distributed UTM in Layered Airspace," in *8th International Conference on Research in Air Transportation*, 2018, pp. 1–8.

[19] K. Dorling, J. Heinrichs, G. G. Messier, and S. Magierowski, "Vehicle Routing Problems for Drone Delivery," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 1, pp. 70–85, jan 2017.

[20] M. F. B. Mohamed Salleh and K. H. Low, "Concept of Operations (ConOps) for Traffic Management of Unmanned Aircraft Systems (TM-UAS) in Urban Environment," in *AIAA Information Systems-AIAA Infotech @ Aerospace*, no. January. Reston, Virginia: American Institute of Aeronautics and Astronautics, jan 2017, pp. 1–13.

[21] J. Krozel, M. Peters, K. D. Bilimoria, C. Lee, and J. S. Mitchell, "System Performance Characteristics of Centralized and Decentralized Air Traffic Separation Strategies," *Air Traffic Control Quarterly*, vol. 9, no. 4, pp. 311–332, oct 2001.

[22] M. Ballin, J. Hoekstra, D. Wing, and G. Lohr, "NASA Langley and NLR Research of Distributed Air/Ground Traffic Management," in *AIAA's Aircraft Technology, Integration, and Operations (ATIO) 2002 Technical Forum*. Reston, Virigina: American Institute of Aeronautics and Astronautics, oct 2002, p. 5826.

[23] J. Klooster, S. Torres, D. Earman, M. Castillo-Effen, R. Subbu, L. Kammer, D. Chan, and T. Tomlinson, "Trajectory synchronization and negotiation in Trajectory Based Operations," in *29th Digital Avionics Systems Conference*. IEEE, oct 2010, pp. 1.A.3–1–1.A.3–11.

[24] K. Wichman, L. Lindberg, L. Kilchert, and O. Bleeker, "Four-Dimensional Trajectory Based Air Traffic Management," in *AIAA Guidance, Navigation, and Con-*

trol Conference and Exhibit. Reston, Virigina: American Institute of Aeronautics and Astronautics, aug 2004, p. 5413.

[25] D.-S. Jang, C. A. Ippolito, S. Sankararaman, and V. Stepanyan, "Concepts of Airspace Structures and System Analysis for UAS Traffic flows for Urban Areas," in *AIAA Information Systems-AIAA Infotech @ Aerospace*, no. January. Reston, Virginia: American Institute of Aeronautics and Astronautics, jan 2017, p. 0449.

[26] E. Sunil, J. Ellerbroek, J. Hoekstra, A. Vidosavljevic, M. Arntzen, F. Bussink, and D. Nieuwenhuisen, "Analysis of Airspace Structure and Capacity for Decentralized Separation Using Fast-Time Simulations," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 1, pp. 38–51, 2017.

[27] S. Bae, H.-S. Shin, C.-H. Lee, and A. Tsourdos, "A New Multiple Flights Routing and Scheduling Algorithm in Terminal Manoeuvring Area," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*. IEEE, sep 2018, pp. 1–9.

[28] S. Bae, H.-S. Shin, and A. Tsourdos, "A New Graph-Based Flight Planning Algorithm for Unmanned Aircraft System Traffic Management," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*. IEEE, sep 2018, pp. 1–9.

[29] S. Bortoff, "Path planning for UAVs," *Proceedings of the 2000 American Control Conference. ACC (IEEE Cat. No.00CH36334)*, no. June, pp. 364–368 vol.1, 2000.

[30] J. Enright, E. Frazzoli, K. Savla, and F. Bullo, "On Multiple UAV Routing with Stochastic Targets: Performance Bounds and Algorithms," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, no. August. Reston, Virigina: American Institute of Aeronautics and Astronautics, aug 2005, p. 5830.

[31] G. B. Lamont, J. N. Slear, and K. Melendez, "UAV swarm mission planning and routing using multi-objective evolutionary algorithms," *IEEE Symposium Computational Intelligence in Multicriteria Decision Making*, no. Mcdm, pp. 10–20, 2007.

[32] J. Hall and D. Anderson, "Reactive route selection from pre-calculated trajectories – application to micro-UAV path planning," *The Aeronautical Journal*, vol. 115, no. 1172, pp. 635–640, oct 2011.

[33] L. De Filippis, G. Guglieri, and F. Quagliotti, "Path Planning Strategies for UAVS in 3D Environments," *Journal of Intelligent & Robotic Systems*, vol. 65, no. 1-4, pp. 247–264, jan 2012.

[34] W. Liu, Z. Zheng, and K.-Y. Cai, "Bi-level programming based real-time path planning for unmanned aerial vehicles," *Knowledge-Based Systems*, vol. 44, pp. 34–47, may 2013.

[35] A. Altmann, M. Niendorf, M. Bednar, and R. Reichel, "Improved 3D Interpolation-Based Path Planning for a Fixed-Wing Unmanned Aircraft," *Journal of Intelligent & Robotic Systems*, vol. 76, no. 1, pp. 185–197, sep 2014.

[36] M. Radmanesh and M. Kumar, "Flight formation of UAVs in presence of moving obstacles using fast-dynamic mixed integer linear programming," *Aerospace Science and Technology*, vol. 50, pp. 149–160, mar 2016.

[37] S. G. Manyam, S. Rasmussen, D. W. Casbeer, K. Kalyanam, and S. Manickam, "Multi-UAV routing for persistent intelligence surveillance & reconnaissance missions," *2017 International Conference on Unmanned Aircraft Systems, ICUAS 2017*, pp. 573–580, 2017.

[38] K. Yu, A. K. Budhiraja, and P. Tokekar, "Algorithms for Routing of Unmanned Aerial Vehicles with Mobile Recharging Stations," *IEEE International Conference on Robotics and Automation 2018*, pp. 1–5, mar 2017.

[39] K. Dorling, J. Heinrichs, G. G. Messier, and S. Magierowski, "Vehicle Routing Problems for Drone Delivery," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 1, pp. 70–85, 2017.

[40] R. T. Wong, "Combinatorial Optimization: Algorithms and Complexity (Christos H. Papadimitriou and Kenneth Steiglitz)," *SIAM Review*, vol. 25, no. 3, pp. 424–425, jul 1983.

[41] J. Y. Yen, "Finding the K Shortest Loopless Paths in a Network," *Management Science*, vol. 17, no. 11, pp. 712–716, jul 1971.

[42] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, dec 1959.

Chapter 6

Conclusions and Future Works

6.1 Conclusions

This thesis aimed to develop effective and extensible algorithms that make multiple vehicles to be simultaneously routed and scheduled to minimise certain performance index in transport planning problems. The algorithms can be applied to each specific traffic management problem by taking into account its operational factors such as minimum separation requirements, feasible speed ranges, heterogeneous agents, and various operational types.

Firstly, for homogeneous multiple flights arriving at an airport in a Terminal Manoeuvring Area (TMA), the thesis proposed a routing & scheduling algorithm in Chapter 2. The algorithm to minimise flight time is based on a weighted digraph in which the proposed flight time weights are assigned to route segments of the airspace represented as the weighted digraph. Dijkstra's algorithm finds each flight's route and schedule simultaneously: this is one of the main novelties compared with existing approaches. However, as its results are independent of each other, the separation between each aircraft are not satisfied. The algorithm solves the same problem iteratively by updating each graph, where the route and schedule of each flight are shared with each other. Also, the computation of each flight routing and scheduling can be done in parallel, which makes the algorithm efficient. We numerically showed that the proposed algo-

rithm can solve the problem of routing and scheduling multiple arrivals at an airport in the TMA while satisfying the separation requirements.

The algorithm developed in Chapter 2 was extended in Chapter 3 to consider heterogeneous multiple arrivals at multiple airports and leading- and following-aircraft-dependent separation in three-dimensional airspace. We introduced both merging points and crossing points in three-dimensional airspace and, utilised aircraft-category-dependent performance data (e.g., the Wake Turbulence Category) required for determination of separation requirements and feasible speed ranges. Also, we considered the case where there are one or more airports in the TMA, and an arrival airport for each aircraft is predetermined while an arrival sequence is not. The decision-making objective is to find the airport that minimises the gap between the last aircraft arrival time and own-ship aircraft arrival time, which reflects runway throughput of each airport. We numerically showed that the proposed algorithm finds near-optimal solution compared with the exhaustive search algorithm, and is scalable with respect to the number of aircraft. Additionally, the case study of the LTMA showed that the proposed algorithm could provide a route and separation-compliant speed profile for each flight. We expect that the proposed algorithm may reduce the ATCOs' workload of managing separation between aircraft by considering separation in flight planning phases.

The algorithms developed in Chapter 2 and Chapter 3 was extended in Chapter 4 and Chapter 5, and tested to different drone delivery operation type for the UTM system. The algorithm proposed in Chapter 4 can support the routing and scheduling of multiple sUASs, in which the entire flight phases are considered within a given urban airspace. Numerical experiments showed that the proposed algorithms provide separation-compliant routes and schedules of multiple sUASs for the first-mile delivery and the last-mile delivery services for two different departure sequencing algorithms, where a single retail point was assumed. It is found that the efficiency of operation is determined by the sequencing algorithms in which the FCFS algorithm and the LCFS algorithm have been applied. From the numerical experiment results, it was suggested to utilise the LCFS algorithm for the last-mile delivery service and the FCFS algorithm for the first-mile delivery service, respectively.

Finally, Chapter 5 conducted a Monte Carlo simulation with 100 runs to analyse

the capacity for the urban airspace. Two different sequencing algorithms, the FCFS algorithm and LCFS algorithm, were compared for four different operation types (1-to-$\mathcal{M}$, $\mathcal{M}$-to-1, $\mathcal{N}$-to-$\mathcal{M}$, and $\mathcal{M}$-to-$\mathcal{N}$, where $\mathcal{M} > \mathcal{N}$). Here, four suggested metrics were utilised. The results for each operation type showed the pros and cons of both sequencing algorithms, and we expect that the results could give many supports to policymakers, urban airspace designers, and regulators.

In summary, this research has developed routing and scheduling algorithms for multiple agent systems. We proposed the algorithm for homogeneous multiple aircraft arriving at a single airport for the ATM system, and then, based on the algorithm, we addressed the algorithms that are applicable to the heterogeneous multiple aircraft arriving at multiple airports for the ATM system. Each flight routing and scheduling problem is solved in parallel based on flight time data. The algorithms can fill the research gap by taking into account spatial-temporal separation that is separately and sequentially considered in the current flight planning. It is expected that the ATCOs' workload to manage the separation between aircraft will be reduced. Also, the algorithms were implemented to the flight planning problems of the UTM system. Meanwhile, we utilised the algorithms to analyse the urban airspace capacity based on the proposed intuitive metrics.

6.2 Future Work

Whilst the proposed algorithms could be promising for the route network-based planning of the transportation systems, there are several remaining issues to be addressed before the algorithms can be utilised as either decision support tools or analysis tools. The following issues could be considered for further research:

1. *Optimality and scalability*: In Chapter 3 we numerically showed the optimality and scalability of the proposed algorithm with regard to the number of agents using the example of a simple route network. However, in order to implement the algorithms on more complex route networks, we need to analytically investigate the optimality with respect to the route network complexity. Also, the analysis

of the scalability shown in the chapter only considered a specific type of aircraft, which should be varied.

2. *Allowing new agents to enter the route network*: The proposed algorithms should consider new agents entering the route network in order to utilise these as a real-time decision support tool. In the ATM system, for example, the proposed algorithms should consider new flights entering the TMA about 30-40 minutes before touchdown. To tackle the impact of this entering aircraft, a study may be conducted following the proposed algorithms. The number of aircraft entering the TMA is controlled by the air traffic flow management system which works with characteristic times closely related to the computational time of the proposed algorithms. The number of flights in the TMA should be planned with a maximum value. The proposed algorithms can be extended to allow new entering flights only if there are an acceptable number of flights the algorithms can take.

3. *Including uncertainties*: Uncertainties produced by external sources such as wind, adverse weather, and unexpected situations have been discussed because it is directly related to safety. To consider the effect of such uncertainties, in general, conservative safety requirements can be implemented such as longer separation requirements between agents. Additional work can improve the efficiency of the transportation system by considering the wind effect where it can be approximated as quasi-static for the time-frame of the flight in the route network. Then, wind effect can be considered to the proposed algorithms to update the computations of the routing and scheduling problem.

4. *Considering both arrival and departure for the ATM system*: For the ATM system, in general, arrival planning and departure planning are considered separately. However, the arrival flights must take into account departure flights for a more efficient operation, especially when there is only one runway. Further research can apply both arrival and departure planning by allocating a predetermined departure sequence and finding an arrival schedule based on the departure sequence.

5. *A better sequencing algorithm*: More studies are needed to develop or to utilise better algorithms for determining the sequence of flights rather than simply utilising the FCFS algorithm in order to minimise the flight time.

6. *Convergence*: Due to the complexity of the problem itself, it is hard to say that there is always a solution. Of course, convergence should be studied through further study. However, it is not easy to analyse convergence analytically because of operational constraints of the systems, so it is possible to solve this by using the characteristics of each system. For example, in ATM system, holding track is used to handle crowded traffic condition at peak time. Further research might find a way to guarantee convergence if holding track is applied to the proposed algorithm.

Appendix A

Numerical Trajectory Optimisation

In order to validate the proposed algorithm applied to the ATM application, we utilise the numerical trajectory optimisation to show that the proposed algorithm can provide feasible flight routes and schedules for ATCO's DST. In this appendix we briefly summarise a trajectory optimisation problem formulation and numerical method utilised in this study.

A.1 Problem Modelling

Dynamic Constraints

In this study, the horizontal motion of the aircraft are expressed by the following set of differential-algebraic equations [1]:

$$\dot{\theta}(t) = \frac{V(t)\sin\psi(t) + W_\theta(\theta(t), \lambda(t))}{r} \tag{A.1}$$

$$\dot{\lambda}(t) = \frac{V(t)\cos\psi(t) + W_\lambda(\theta(t), \lambda(t))}{r\cos\theta(t)} \tag{A.2}$$

$$\dot{V}(t) = \frac{T(t) - D(V(t), C_L(t))}{m(t)} \tag{A.3}$$

$$\dot{\psi}(t) = \frac{L(V(t), m(t)) \sin \phi(t)}{m(t)V(t)} \tag{A.4}$$

$$\dot{m}(t) = -T(t)\eta(V(t)) \tag{A.5}$$

where the state vector is $x(t) = (\theta(t), \lambda(t), V(t), \psi(t), m(t))$. θ, λ, V, ψ and m denote longitude, latitude, TAS, heading angle, and aircraft mass, respectively. The control vector is $u(t) = (T(t), C_L, \phi(t))$, where T, C_L and ϕ are the engine thrust, lift coefficient, and bank angle, respectively. W_θ and W_λ denote the components of the wind vector. g and r are the acceleration of gravity and Earth radius, which are assumed as constant. The specific fuel consumption corresponds to η. D and L are aerodynamic drag and lift, respectively. The aerodynamic drag is proportional to the airspeed [2]:

$$D = \frac{1}{2}\rho S V^2 C_D \tag{A.6}$$

where C_D is the drag coefficient, ρ is the density of the air and S is the wing area. In this model, the drag coefficient is expressed as a polynomial function of the lift coefficient (C_L), which is one of the control inputs. The aerodynamic lift is:

$$L = \frac{1}{2}\rho S V^2 C_L. \tag{A.7}$$

Path Constraints

We utilise the BADA v3, which provides models for fuel consumption, thrust, aerodynamic force, performance limitations, etc. The aircraft motions are constrained by performance limits. The performance constraints are considered to define the domain of state and control variables.

$$m_{\min} \leq \quad m(t) \quad \leq m_{\max} \tag{A.8}$$

$$T_{\min} \leq \quad T(t) \quad \leq T_{\max} \tag{A.9}$$

$$C_{V_{\min}} V_{\text{stall}} \leq \quad V(t) \quad \leq V_{\max} \tag{A.10}$$

$$0 \leq \quad C_L(t) \quad \leq C_{L_{\max}} \tag{A.11}$$

$$\phi_{\min,\text{civ}} \leq \quad \phi(t) \quad \leq \phi_{\max,\text{civ}} \tag{A.12}$$

$$\dot{V}(t) \quad \leq \dot{a}_{1,\max,\text{civ}} \tag{A.13}$$

More details can be found in the BADA database manual [3]

Meteorological Model

Wind forecast data are provided by the National Oceanic and Atmospheric Administration to take into account the influence of wind. The wind forecast data are fitted into analytic functions, in which the functions are 4$^{\text{th}}$ order multiple linear regressions [4].

$$W_\theta = \beta_{00}^\theta + \beta_{10}^\theta\theta + \beta_{01}^\theta\lambda + \beta_{20}^\theta\theta^2 + \cdots + \beta_{13}^\theta\theta\lambda^3 + \beta_{04}^\theta\lambda^4 \tag{A.14}$$

$$W_\lambda = \beta_{00}^\lambda + \beta_{10}^\lambda\theta + \beta_{01}^\lambda\lambda + \beta_{20}^\lambda\theta^2 + \cdots + \beta_{13}^\lambda\theta\lambda^3 + \beta_{04}^\lambda\lambda^4 \tag{A.15}$$

Performance Index

We utilise the Direct Operating Cost (DOC), which is decided by airliners and aircraft owners, as an objective function. The DOC is composed of the fuel and time cost. The performance index can be written as follows:

$$J(\text{DOC}) = \int_{t_0}^{t_f} [FF(x,u) + \text{CI}]dt \tag{A.16}$$

where FF is fuel flow. t_0 and t_f are the initial and terminal time, respectively. The CI, which is an abbreviation of Cost Index, is the ratio of time related cost and the cost of fuel. When the CI is equal to zero, it generates the fuel optimal trajectory, in the contrast, as the CI increases, the time performance index tends to increasing [5].

A.2 Optimal Control Problem

Continuous Bolza Problem

The aircraft trajectory optimisation problem is often formulated as a multi-phase constrained Optimal Control Problem (OCP), in which the objective of the OCP is to find the state and control that minimises or maximises an objective function such that constrains for dynamics, path constraints, and boundaries. For some simple problems, its solution can be obtained analytically from the necessary and sufficient conditions of optimality

The OCP can be formulated in the Bolza form as follows:

$$J = \Phi(\mathbf{x}(t_0), t_0, \mathbf{x}(t_f), t_f) + \int_{t_0}^{t_f} L(\mathbf{x}(t), \mathbf{u}(t))dt, \qquad t \in [t_0, t_f] \tag{A.17}$$

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \qquad t \in [t_0, t_f] \tag{A.18}$$

$$\mathbf{C}(\mathbf{x}(t), \mathbf{u}(t)) \leq \mathbf{0}, \qquad t \in [t_0, t_f] \tag{A.19}$$

$$\varphi(\mathbf{x}(t_0), t_0, \mathbf{x}(t_f), t_f) = \mathbf{0} \tag{A.20}$$

where $\mathbf{x}(t) \in \mathbb{R}^{\mathrm{n}}$, $\mathbf{u}(t) \in \mathbb{R}^{\mathrm{m}}$, and t are the state, control input and time, respectively. To simplify the problem, the OCP can be divided into K mesh intervals, $t_0 < t_1 < t_2 < \cdots t_K = t_f$ on the interval $t \in [t_0, t_f]$. In each interval k, called *mesh interval*, $t \in [t_{k-1}, t_k]$ can be transformed into $\tau \in [-1, +1]$ via the affine transformation:

$$\tau = \frac{2t - (t_{k-1} + t_k)}{t_k - t_{k-1}} \tag{A.21}$$

which yields

$$\frac{dt}{d\tau} = \frac{t_k - t_{k-1}}{2}, \qquad (k = 1, \cdots, K) \tag{A.22}$$

Next, $\mathbf{x}^{(k)}(\tau)$ and $\mathbf{u}^{(k)}(\tau)$ are supposed to be the state and control input in the k^{th} mesh interval. Then the original OCP of Equations (A.17)−(A.20) can be reformulated in terms of the defined variables as follows. Minimise the cost functional

$$J = \Phi(\mathbf{x}^{(1)}(-1), t_0, \mathbf{x}^{(K)}(1), t_K) + \sum_{k=1}^{K} \frac{t_k - t_{k-1}}{2} \int_{-1}^{1} \mathbf{L}(\mathbf{x}^{(k)}(t), \mathbf{u}^{(k)}(t), \tau; t_{k-1}, t_k) d\tau, \qquad t \in [t_0, t_f] \tag{A.23}$$

subject to

$$\frac{d\mathbf{x}^{(k)}(\tau)}{d\tau} = \frac{t_k - t_{k-1}}{2} \mathbf{f}(\mathbf{x}^{(k)}(\tau), \mathbf{u}^{(k)}(\tau), \tau; t_{k-1}, t_k), \qquad (k = 1, \cdots, K) \tag{A.24}$$

$$\mathbf{C}(\mathbf{x}^{(k)}(\tau), \mathbf{u}^{(k)}(\tau), \tau; t_{k-1}, t_k) \leq \mathbf{0}, \qquad (k = 1, \cdots, K) \tag{A.25}$$

$$\varphi(\mathbf{x}^{(1)}(-1), t_0, \mathbf{x}^{(K)}(1), t_K) = \mathbf{0} \tag{A.26}$$

and the interior point constraints:

$$\mathbf{x}^{(k)}(1) - \mathbf{x}^{(k+1)}(-1) = 0, \qquad (k = 1, \cdots, K-1) \tag{A.27}$$

Radau Pseudospectral Method (RPM)

Now let us describe the parametrisation of the OCP. In this research, a general purpose optimal control software package, GPOPS-II, is used for solving the re-exit trajectory optimisation problem [6]. RPM is one of the advanced variable order Gaussian quadrature collocation methods, and is implemented in GPOPS-II. In the RPM approximation of the OCP, the infinite-dimensional OCP is converted into a finite-dimensional NLP problem. GPOPS-II utilises hp-adaptive methods that allow to vary the number of mesh intervals and the degree of the approximating polynomial in each mesh interval. The RPM uses the Lagrange polynomial approximation at a set of discrete Legendre-Gauss-Radau (LGR) collocation points. In each phase, the state can be approximated by the

following polynomial:

$$\mathbf{x}^{(k)}(\tau) \approx \mathbf{X}^{(k)}(\tau) = \sum_{j=0}^{N_k} \mathbf{X}_j^{(k)} \mathbf{L}_j^{(k)}(\tau) \tag{A.28}$$

where $\mathbf{L}_j^{(k)}(\tau)$ are defined as:

$$\mathbf{L}_j^{(k)}(\tau) = \prod_{i=0, i\neq j}^{N_k} \frac{\tau - \tau_i^{(k)}}{\tau_j^{(k)} - \tau_i^{(k)}}, \qquad j \in \{0, 1, \cdots, N_k\} \tag{A.29}$$

where $\tau \in [-1, +1]$, $(\tau_1^{(k)}, \cdots, \tau_{N_k}^{(k)})$ are the LGR collocation points in the k^{th} mesh interval and the final point $\tau_{N_k}^{(k)}$ is a non-collocation point. The control can be also approximated as follows:

$$\mathbf{u}^{(k)}(\tau) \approx \mathbf{U}^{(k)}(\tau) = \sum_{j=1}^{N_k} \mathbf{U}_j^{(k)} \tilde{\mathbf{L}}_j^{(k)}(\tau) \tag{A.30}$$

where $\tilde{\mathbf{L}}_j^{(k)}(\tau)$ are defined as:

$$\tilde{\mathbf{L}}_j^{(k)}(\tau) = \prod_{i=1, i\neq j}^{N_k} \frac{\tau - \tau_i^{(k)}}{\tau_j^{(k)} - \tau_i^{(k)}}, \qquad j \in \{1, \cdots, N_k\} \tag{A.31}$$

The Lagrange polynomials Eq.(A.29) and Eq.(A.31) have the property:

$$\mathbf{L}_j^{(k)}\left(\tau_i^{(k)}\right) = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \tag{A.32}$$

$$\tilde{\mathbf{L}}_j^{(k)}\left(\tau_i^{(k)}\right) = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \tag{A.33}$$

Differentiating the expression in Equation (A.28) w.r.t. τ produces:

$$\dot{\mathbf{X}}^{(k)}(\tau) \equiv \frac{d\mathbf{X}^{(k)}(\tau)}{d\tau} = \sum_{j=0}^{N_k} \mathbf{X}_j^{(k)} \dot{\mathbf{L}}_j^{(k)}(\tau) = \sum_{j=0}^{N_k} \mathbf{X}_j^{(k)} \mathbf{D}_{ij}^{(k)} \tag{A.34}$$

$$\mathbf{D}_{ij}^{(k)} = \dot{\mathbf{L}}_j^{(k)}(\tau_i^{(k)}), \quad \text{for } i = 1, \cdots N_k,\ j = 0, 1, \cdots N_k \tag{A.35}$$

Transcribed algebraic constraints of the dynamic constraints via the differential approximation matrix are as follows:

$$\sum_{j=0}^{N_k} \mathbf{D}_{ij}^{(k)} \mathbf{X}_j^{(k)} = \frac{t_k - t_{k-1}}{2} \mathbf{f}(\mathbf{X}_i^{(k)}, \mathbf{U}_i^{(k)}, \tau_i^{(k)}; t_{k-1}, t_k), \quad \text{for } i = 1, \cdots N_k \tag{A.36}$$

Furthermore, the inequality path constraints can be calculated at the N_k points in each mesh interval as follows:

$$\mathbf{C}(\mathbf{X}_i^{(k)}, \mathbf{U}_i^{(k)}, \tau_i^{(k)}, t_{k-1}, t_k) \leq \mathbf{0}, \quad \text{for } i = 1, \cdots, N_k \tag{A.37}$$

Finally, the boundary condition can be rewritten at the N_k points in each mesh interval as:

$$\varphi(\mathbf{X}_0^{(1)}, t_0, \mathbf{X}_{N_k}^{(K)}, t_K) = \mathbf{0} \tag{A.38}$$

The continuous-time cost functional can be constituted by the multi-interval at LGR points, resulting in:

$$J \cong \Phi(\mathbf{X}_0^{(1)}, t_0, \mathbf{X}_{N_k}^{(K)}, t_K) + \sum_{k=1}^{K} \sum_{j=0}^{N_k} \frac{t_k - t_{k-1}}{2} \omega_j \mathbf{L}(\mathbf{X}_j^{(k)}, \mathbf{U}_j^{(k)}, \tau_j^{(k)}; t_{k-1}, t_k) \tag{A.39}$$

where ω_j represent the LGR weights. Then, the NLP problem is used to minimise Equation (A.39) subject to Equation (A.34)$-$(A.38). Then the existing nonlinear programming solver Interior Point OPTimizer (IPOPT) can be applied to the NLP problem using hp-adaptive methods [6, 7].

References

[1] D. G. Hull, *Fundamentals of airplane flight mechanics*, 2007.

[2] J. Roskam and C.-T. E. Lan, *Airplane aerodynamics and performance*. DARcorporation, 1997.

[3] Eurocontrol, "User Manual for the Base of Aircraft Data (Bada) Revision 3.12," Eurocontrol, Tech. Rep. August, 2014.

[4] M. Fernando and S. Arnedo, *Commercial Aircraft Trajectory Planning based on Multiphase Mixed-Integer Optimal Control by*, 2013, no. April.

[5] R. Bill, "Fuel Conservation Strategies: Cost Index Explained," *Boeing Aero Magazine*, pp. 26–28, 2007.

[6] M. A. Patterson and A. V. Rao, "GPOPS-II," *ACM Transactions on Mathematical Software*, vol. 41, no. 1, pp. 1–37, oct 2014. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2684421.2558904

[7] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, mar 2006. [Online]. Available: http://link.springer.com/10.1007/s10107-004-0559-y

Bibliography

References for Chapter 1

[1] H. Balakrishnan and B. Chandran, "Scheduling Aircraft Landings Under Constrained Position Shifting," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*. Reston, Virigina: American Institute of Aeronautics and Astronautics, aug 2006, p. 6320.

[2] D. R. Isaacson, a. V. Sadovsky, and D. Davis, "Tactical Scheduling for Precision Air Traffic Operations: Past Research and Current Problems." *Journal of Aerospace Information Systems*, vol. 11, no. 4, pp. 234–257, 2014.

[3] H. Fazlollahtabar and M. Saidi-Mehrabad, "Methodologies to Optimize Automated Guided Vehicle Scheduling and Routing Problems: A Review Study," *Journal of Intelligent & Robotic Systems*, vol. 77, no. 3-4, pp. 525–545, 2015.

[4] R. Breil, D. Delahaye, L. Lapasset, and E. Feron, "Multi-agent systems for air traffic conflicts resolution by local speed regulation and departure delay," in *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, 2016, pp. 1–10.

[5] Hanbong Lee and H. Balakrishnan, "A Study of Tradeoffs in Scheduling Terminal-Area Operations," *Proceedings of the IEEE*, vol. 96, no. 12, pp. 2081–2095, dec 2008.

[6] A. V. Sadovsky, "Application of the Shortest-Path Problem to Routing Terminal Airspace Air Traffic," *Journal of Aerospace Information Systems*, vol. 11, no. 3, pp. 118–130, mar 2014.

[7] A. V. Sadovsky, D. Davis, and D. R. Isaacson, "Efficient Computation of Separation-Compliant Speed Advisories for Air Traffic Arriving in Terminal Airspace," *Journal of Dynamic Systems, Measurement, and Control*, vol. 136, no. 4, p. 041027, may 2014.

[8] A. Rezaei, A. V. Sadovsky, and J. L. Speyer, "Existence and Determination of Separation-Compliant Speed Control in Terminal Airspace," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 6, pp. 1374–1391, jun 2016.

[9] M. F. B. Mohamed Salleh, C. Wanchao, Z. Wang, S. Huang, D. Y. Tan, T. Huang, and K. H. Low, "Preliminary Concept of Adaptive Urban Airspace Management for Unmanned Aircraft Operations," in *2018 AIAA Information Systems-AIAA Infotech @ Aerospace*, no. January. Reston, Virginia: American Institute of Aeronautics and Astronautics, jan 2018, pp. 1–12.

[10] D. R. Isaacson, a. V. Sadovsky, and D. Davis, "Tactical Scheduling for Precision Air Traffic Operations: Past Research and Current Problems." *Journal of Aerospace Information Systems*, vol. 11, no. 4, pp. 234–257, 2014.

[11] S. Bradford, "Version 1.0 of the Unmanned Aireraft Systems (UAS) Traffic Management (UTM) Concept of Operations," Federal Aviation Administration, Tech. Rep., 2018.

[12] T. Prevot, J. Homola, and J. Mercer, "From Rural to Urban Environments: Human/Systems Simulation Research for Low Altitude UAS Traffic Management (UTM)," in *16th AIAA Aviation Technology, Integration, and Operations Conference*. Reston, Virginia: American Institute of Aeronautics and Astronautics, jun 2016, p. 3291.

[13] Geister and Dagi, "Concept for Urban Airspace Integration DLR U-Space Blueprint," Institute of Flight Guidance, Tech. Rep. December, 2017. [Online]. Available: http://www.dlr.de/fl/desktopdefault.aspx/tabid-11763/20624{_}read-48305/

[14] C. A. CASTIGLIONI, A. S. RABUFFETTI, G. P. Chiarelli, G. BRAMBILLA, and J. Georgi, "Unmanned aerial vehicle (UAV) application to the structural health assessment of large civil engineering structures," in *Fifth International Conference on Remote Sensing and Geoinformation of the Environment (RSCy2017)*, G. Papadavid, D. G. Hadjimitsis, S. Michaelides, V. Ambrosia, K. Themistocleous, and G. Schreier, Eds. SPIE, sep 2017, p. 29.

[15] M. Cannioto, A. D'Alessandro, G. Lo Bosco, S. Scudero, and G. Vitale, "Brief communication: Vehicle routing problem and UAV application in the post-earthquake scenario," *Natural Hazards and Earth System Sciences*, vol. 17, no. 11, pp. 1939–1946, nov 2017.

[16] Eurocontrol, "Demand Data Repository 2," 2017. [Online]. Available: https://www.eurocontrol.int/articles/ddr2-web-portal

References for Chapter 2

[1] SESAR Consortium, "European ATM Master Plan - Edition 2," *The Roadmap for Sustainable Air Traffic Management*, no. October, pp. 1–100, 2012.

[2] Federal Aviation Administration, "NextGen Update: 2014," Federal Aviation Administration, Tech. Rep. August, 2014.

[3] G. C. Carr, H. Erzberger, and F. Neuman, "Delay Exchanges in Arrival Sequencing and Scheduling," *Journal of Aircraft*, vol. 36, no. 5, pp. 785–791, sep 1999.

[4] A. Bayen, C. Tomlin, Yinyu Ye, and Jiawei Zhang, "An approximation algorithm for scheduling aircraft with holding time," in *2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601)*. IEEE, 2004, pp. 2760–2767 Vol.3.

[5] L. Bianco, P. Dell'Olmo, and S. Giordani, "Scheduling models for air traffic control in terminal areas," *Journal of Scheduling*, vol. 9, no. 3, pp. 223–253, jun 2006.

[6] H. Balakrishnan and B. Chandran, "Scheduling Aircraft Landings Under Constrained Position Shifting," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*. Reston, Virigina: American Institute of Aeronautics and Astronautics, aug 2006, p. 6320.

[7] Hanbong Lee and H. Balakrishnan, "A Study of Tradeoffs in Scheduling Terminal-Area Operations," *Proceedings of the IEEE*, vol. 96, no. 12, pp. 2081–2095, dec 2008.

[8] R. Chipalkatty, P. Twu, A. Rahmani, and M. Egerstedt, "Distributed scheduling for air traffic throughput maximization during the terminal phase of flight," *Proceedings of the IEEE Conference on Decision and Control*, pp. 1195–1200, 2010.

[9] A. V. Sadovsky, D. Davis, and D. R. Isaacson, "Separation-compliant, optimal routing and control of scheduled arrivals in a terminal airspace," *Transportation Research Part C: Emerging Technologies*, vol. 37, pp. 157–176, 2013.

[10] D. R. Isaacson, a. V. Sadovsky, and D. Davis, "Tactical Scheduling for Precision Air Traffic Operations: Past Research and Current Problems." *Journal of Aerospace Information Systems*, vol. 11, no. 4, pp. 234–257, 2014.

[11] A. V. Sadovsky, "Application of the Shortest-Path Problem to Routing Terminal Airspace Air Traffic," *Journal of Aerospace Information Systems*, vol. 11, no. 3, pp. 118–130, mar 2014.

[12] A. V. Sadovsky and M. M. Jastrzebski, "Strategic time-based metering that assures separation for integrated operations in a terminal airspace," *15th AIAA Aviation Technology, Integration, and Operations Conference*, no. June, pp. 1–14, 2015.

[13] R. Breil, D. Delahaye, L. Lapasset, and E. Feron, "Multi-agent systems for air traffic conflicts resolution by local speed regulation and departure delay," in *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, 2016, pp. 1–10.

[14] A. Rezaei, A. V. Sadovsky, and J. L. Speyer, "Existence and Determination of Separation-Compliant Speed Control in Terminal Airspace," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 6, pp. 1374–1391, jun 2016.

[15] S. Sidiropoulos, K. Han, A. Majumdar, and W. Y. Ochieng, "Robust identification of air traffic flow patterns in Metroplex terminal areas under demand uncertainty," *Transportation Research Part C: Emerging Technologies*, vol. 75, pp. 212–227, 2017.

[16] M. Samà, A. D'Ariano, P. D'Ariano, and D. Pacciarelli, "Scheduling models for optimal aircraft traffic control at busy airports: Tardiness, priorities, equity and violations considerations," *Omega (United Kingdom)*, vol. 67, pp. 81–98, 2017.

[17] A. Bayen, C. Tomlin, Yinyu Ye, and Jiawei Zhang, "MILP formulation and polynomial time algorithm for an aircraft scheduling problem," in *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*. IEEE, 2003, pp. 5003–5010.

[18] R. T. Wong, "Combinatorial Optimization: Algorithms and Complexity (Christos H. Papadimitriou and Kenneth Steiglitz)," *SIAM Review*, vol. 25, no. 3, pp. 424–425, jul 1983.

[19] V. V. Vazirani, *Approximation Algorithms*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003.

[20] H. Chida, C. Zuniga, and D. Delahaye, "Topology design for integrating and sequencing flows in terminal maneuvering area," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 230, no. 9, pp. 1705–1720, jul 2016.

[21] Eurocontrol, "Eurocontrol TBS Project: Aircraft Wake Vortex Modelling in support of the Time Based Separation Project, AO-06-11079EB," Eurocontrol, Tech. Rep., 2008.

[22] Eurocontrol, "Eurocontrol TBS Project: Results from the Decenber 2007 Time Based Separation Real-Time Simulation Exercises, EEC Report No. 411," Eurocontrol, Tech. Rep., 2009.

[23] C. Morris, J. Peters, and P. Choroba, "Validation of the time based separation concept at london heathrow airport," *Proceedings of the 10th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2013*, 2013.

[24] G. R. van Brummelen, *Heavenly Mathematics: The Forgotten Art of Spherical Trigonometry*, mar 2015.

[25] International Civil Aviation Organization, "Aircraft Operations. Volume I - Flight Procedures," International Civil Aviation Organization, Tech. Rep. October, 2006.

[26] Eurocontrol, "User Manual for the Base of Aircraft Data (Bada) Revision 3.12," Eurocontrol, Tech. Rep. August, 2014.

[27] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, dec 1959.

[28] M. Mesgarpour, C. N. Potts, and J. A. Bennell, "Models for aircraft landing optimization," *Proceedings of the 4th international conference on research in air transportation (ICRAT2010)*, pp. 1–4, 2010.

[29] Eurocontrol, "Demand Data Repository 2," 2017. [Online]. Available: https://www.eurocontrol.int/articles/ddr2-web-portal

[30] National Air Traffic Services, "Aeronautical Information Service," 2017. [Online]. Available: https://www.nats.aero/do-it-online/ais/

References for Chapter 3

[1] Eurocontrol, "Demand Data Repository 2," 2017. [Online]. Available: https://www.eurocontrol.int/articles/ddr2-web-portal

[2] A. V. Sadovsky, D. Davis, and D. R. Isaacson, "Efficient Computation of Separation-Compliant Speed Advisories for Air Traffic Arriving in Terminal Airspace," *Journal of Dynamic Systems, Measurement, and Control*, vol. 136, no. 4, p. 041027, may 2014.

[3] A. V. Sadovsky and M. M. Jastrzebski, "Strategic time-based metering that assures separation for integrated operations in a terminal airspace," *15th AIAA Aviation Technology, Integration, and Operations Conference*, no. June, pp. 1–14, 2015.

[4] A. V. Sadovsky, "Application of the Shortest-Path Problem to Routing Terminal Airspace Air Traffic," *Journal of Aerospace Information Systems*, vol. 11, no. 3, pp. 118–130, mar 2014.

[5] A. Rezaei, A. V. Sadovsky, and J. L. Speyer, "Existence and Determination of Separation-Compliant Speed Control in Terminal Airspace," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 6, pp. 1374–1391, jun 2016.

[6] A. V. Sadovsky, D. Davis, and D. R. Isaacson, "Separation-compliant, optimal routing and control of scheduled arrivals in a terminal airspace," *Transportation Research Part C: Emerging Technologies*, vol. 37, pp. 157–176, 2013.

[7] D. R. Isaacson, a. V. Sadovsky, and D. Davis, "Tactical Scheduling for Precision Air Traffic Operations: Past Research and Current Problems." *Journal of Aerospace Information Systems*, vol. 11, no. 4, pp. 234–257, 2014.

[8] R. Breil, D. Delahaye, L. Lapasset, and E. Feron, "Multi-agent systems for air traffic conflicts resolution by local speed regulation and departure delay," in *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, 2016, pp. 1–10.

[9] S. Sidiropoulos, K. Han, A. Majumdar, and W. Y. Ochieng, "Robust identification of air traffic flow patterns in Metroplex terminal areas under demand uncertainty," *Transportation Research Part C: Emerging Technologies*, vol. 75, pp. 212–227, 2017.

[10] S. Bae, H.-S. Shin, C.-H. Lee, and A. Tsourdos, "A New Multiple Flights Routing and Scheduling Algorithm in Terminal Manoeuvring Area," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*. IEEE, sep 2018, pp. 1–9.

[11] S. Bae, H.-S. Shin, and A. Tsourdos, "A New Graph-Based Flight Planning Algorithm for Unmanned Aircraft System Traffic Management," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*. IEEE, sep 2018, pp. 1–9.

[12] R. T. Wong, "Combinatorial Optimization: Algorithms and Complexity (Christos H. Papadimitriou and Kenneth Steiglitz)," *SIAM Review*, vol. 25, no. 3, pp. 424–425, jul 1983.

[13] Eurocontrol, "Eurocontrol TBS Project: Aircraft Wake Vortex Modelling in support of the Time Based Separation Project, AO-06-11079EB," Eurocontrol, Tech. Rep., 2008.

[14] Eurocontrol, "Eurocontrol TBS Project: Results from the Decenber 2007 Time Based Separation Real-Time Simulation Exercises, EEC Report No. 411," Eurocontrol, Tech. Rep., 2009.

[15] C. Morris, J. Peters, and P. Choroba, "Validation of the time based separation concept at london heathrow airport," *Proceedings of the 10th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2013*, 2013.

[16] International Civil Aviation Organization, "Aircraft Operations. Volume I - Flight Procedures," International Civil Aviation Organization, Tech. Rep. October, 2006.

[17] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, dec 1959.

[18] M. Mesgarpour, C. N. Potts, and J. A. Bennell, "Models for aircraft landing optimization," *Proceedings of the 4th international conference on research in air transportation (ICRAT2010)*, pp. 1–4, 2010.

[19] National Air Traffic Services, "Aeronautical Information Service," 2017. [Online]. Available: https://www.nats.aero/do-it-online/ais/

References for Chapter 4

[1] Amazon.com Inc., "Determining Safe Access with a Best- Equipped, Best-Served Model for Small Unmanned Aircraft Systems," *NASA UTM 2015: The Next Era of Aviation*, 2015.

[2] C. A. Thiels, J. M. Aho, S. P. Zietlow, and D. H. Jenkins, "Use of unmanned aerial vehicles for medical product transport," *Air Medical Journal*, vol. 34, no. 2, pp. 104–108, 2015.

[3] Federal Aviation Administration, "Operation and Certification of Small Unmanned Aircraft Systems," Federal Aviation Administration, Tech. Rep. 124, 2016. [Online]. Available: https://federalregister.gov/a/2016-15079

[4] Geister and Dagi, "Concept for Urban Airspace Integration DLR U-Space Blueprint," Institute of Flight Guidance, Tech. Rep. December, 2017. [Online]. Available: http://www.dlr.de/fl/desktopdefault.aspx/tabid-11763/20624{_}read-48305/

[5] J. Hoekstra, S. Kern, O. Schneider, F. Knabe, and B. Lamiscarre, "Metropolis – Urban Airspace Design," Technical University of Delft National, Tech. Rep., 2015.

[6] J. Rios, D. Mulfinger, J. Homola, and P. Venkatesan, "NASA UAS traffic management national campaign: Operations across Six UAS Test Sites," in *2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)*. IEEE, sep 2016, pp. 1–6.

[7] M. F. B. Mohamed Salleh and K. H. Low, "Concept of Operations (ConOps) for Traffic Management of Unmanned Aircraft Systems (TM-UAS) in Urban Environment," in *AIAA Information Systems-AIAA Infotech @ Aerospace*, no. January. Reston, Virginia: American Institute of Aeronautics and Astronautics, jan 2017, pp. 1–13.

[8] S. Bortoff, "Path planning for UAVs," *Proceedings of the 2000 American Control Conference. ACC (IEEE Cat. No.00CH36334)*, no. June, pp. 364–368 vol.1, 2000.

[9] J. Enright, E. Frazzoli, K. Savla, and F. Bullo, "On Multiple UAV Routing with Stochastic Targets: Performance Bounds and Algorithms," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, no. August. Reston, Virigina: American Institute of Aeronautics and Astronautics, aug 2005, p. 5830.

[10] G. B. Lamont, J. N. Slear, and K. Melendez, "UAV swarm mission planning and routing using multi-objective evolutionary algorithms," *IEEE Symposium Computational Intelligence in Multicriteria Decision Making*, no. Mcdm, pp. 10–20, 2007.

[11] S. G. Manyam, S. Rasmussen, D. W. Casbeer, K. Kalyanam, and S. Manickam, "Multi-UAV routing for persistent intelligence surveillance & reconnaissance missions," *2017 International Conference on Unmanned Aircraft Systems, ICUAS 2017*, pp. 573–580, 2017.

[12] K. Yu, A. K. Budhiraja, and P. Tokekar, "Algorithms for Routing of Unmanned Aerial Vehicles with Mobile Recharging Stations," *IEEE International Conference on Robotics and Automation 2018*, pp. 1–5, mar 2017.

[13] K. Dorling, J. Heinrichs, G. G. Messier, and S. Magierowski, "Vehicle Routing Problems for Drone Delivery," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 1, pp. 70–85, 2017.

[14] M. Samà, A. D'Ariano, P. D'Ariano, and D. Pacciarelli, "Scheduling models for optimal aircraft traffic control at busy airports: Tardiness, priorities, equity and violations considerations," *Omega (United Kingdom)*, vol. 67, pp. 81–98, 2017.

[15] S. Bae, H.-S. Shin, C.-H. Lee, and A. Tsourdos, "A New Multiple Flights Routing and Scheduling Algorithm in Terminal Manoeuvring Area," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*. IEEE, sep 2018, pp. 1–9.

[16] S. Bae, H.-S. Shin, and A. Tsourdos, "A New Graph-Based Flight Planning Algorithm for Unmanned Aircraft System Traffic Management," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*. IEEE, sep 2018, pp. 1–9.

[17] M. F. B. Mohamed Salleh, C. Wanchao, Z. Wang, S. Huang, D. Y. Tan, T. Huang, and K. H. Low, "Preliminary Concept of Adaptive Urban Airspace Management for Unmanned Aircraft Operations," in *2018 AIAA Information Systems-AIAA Infotech @ Aerospace*, no. January. Reston, Virginia: American Institute of Aeronautics and Astronautics, jan 2018, pp. 1–12.

[18] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, dec 1959.

[19] P. Hart, N. 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.

[20] R. T. Wong, "Combinatorial Optimization: Algorithms and Complexity (Christos H. Papadimitriou and Kenneth Steiglitz)," *SIAM Review*, vol. 25, no. 3, pp. 424–425, jul 1983.

[21] J. Y. Yen, "Finding the K Shortest Loopless Paths in a Network," *Management Science*, vol. 17, no. 11, pp. 712–716, jul 1971.

References for Chapter 5

[1] N. Lavars, "Amazon to begin testing new delivery drones in the US," *New Atlas*, 2015.

[2] Amazon.com Inc., "Determining Safe Access with a Best- Equipped, Best-Served Model for Small Unmanned Aircraft Systems," *NASA UTM 2015: The Next Era of Aviation*, 2015.

[3] D. Simmons, "Rwanda begins Zipline commercial drone deliveries," *BBC World News*, 2016.

[4] C. A. Thiels, J. M. Aho, S. P. Zietlow, and D. H. Jenkins, "Use of unmanned aerial vehicles for medical product transport," *Air Medical Journal*, vol. 34, no. 2, pp. 104–108, 2015.

[5] L. A. Haidari, S. T. Brown, M. Ferguson, E. Bancroft, M. Spiker, A. Wilcox, R. Ambikapathi, V. Sampath, D. L. Connor, and B. Y. Lee, "The economic and operational value of using drones to transport vaccines," *Vaccine*, vol. 34, no. 34, pp. 4062–4067, jul 2016.

[6] J. Hoekstra, S. Kern, O. Schneider, F. Knabe, and B. Lamiscarre, "Metropolis – Urban Airspace Design," Technical University of Delft National, Tech. Rep., 2015.

[7] P. Kopardekar, "Unmanned Aerial System (UAS) Traffic Management (UTM): Enabling Low-Altitude Airspace and UAS Operations," National Aeronautics and Space Administration, Tech. Rep. April, 2014.

[8] T. Prevot, J. Rios, P. Kopardekar, J. E. Robinson III, M. Johnson, and J. Jung, "UAS Traffic Management (UTM) Concept of Operations to Safely Enable Low Altitude Flight Operations," *16th AIAA Aviation Technology, Integration, and Operations Conference*, no. June, pp. 1–16, 2016.

[9] V. Bulusu, R. Sengupta, V. Polishchuk, and L. Sedov, "Cooperative and non-cooperative UAS traffic volumes," *2017 International Conference on Unmanned Aircraft Systems, ICUAS 2017*, pp. 1673–1681, 2017.

[10] L. Sedov, V. Polishchuk, and V. Bulusu, "Sampling-based capacity estimation for unmanned traffic management," *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, vol. 2017-Septe, 2017.

[11] M. F. B. Mohamed Salleh, C. Wanchao, Z. Wang, S. Huang, D. Y. Tan, T. Huang, and K. H. Low, "Preliminary Concept of Adaptive Urban Airspace Management for Unmanned Aircraft Operations," in *2018 AIAA Information Systems-AIAA Infotech @ Aerospace*, no. January. Reston, Virginia: American Institute of Aeronautics and Astronautics, jan 2018, pp. 1–12.

[12] I. V. Laudeman, S. G. Shelden, R. Branstrom, and C. L. Brasil, "Dynamic Density: An Air Traffic Management Metric," National Aeronautics and Space Administration, Tech. Rep. April, 1998.

[13] A. Majumdar, W. Ochieng, and J. Polak, "Estimation of European Airspace Capacity from a Model of Controller Workload," *Journal of Navigation*, vol. 55, no. 03, sep 2002.

[14] A. Majumdar, W. Y. Ochieng, J. Bentham, and M. Richards, "En-route sector capacity estimation methodologies: An international survey," *Journal of Air Transport Management*, vol. 11, no. 6, pp. 375–387, nov 2005.

[15] A. Klein, L. Cook, B. Wood, and D. Simenauer, "Airspace capacity estimation using flows and Weather-Impacted Traffic Index," in *2008 Integrated Communications, Navigation and Surveillance Conference*. IEEE, may 2008, pp. 1–12.

[16] A. J. H. Qinetiq, E. Care, and I. Metrics, "Assessing the Capacity of Novel ATM Systems," in *4th USA/Europe Air Traffic Management R&D Seminar*, no. December, 2001.

[17] V. Bulusu, V. Polishchuk, R. Sengupta, and L. Sedov, "Capacity Estimation for Low Altitude Airspace," *17th AIAA Aviation Technology, Integration, and Operations Conference*, no. June, pp. 1–15, 2017.

[18] L. Sedov and V. Polishchuk, "Centralized and Distributed UTM in Layered Airspace," in *8th International Conference on Research in Air Transportation*, 2018, pp. 1–8.

[19] K. Dorling, J. Heinrichs, G. G. Messier, and S. Magierowski, "Vehicle Routing Problems for Drone Delivery," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 1, pp. 70–85, jan 2017.

[20] M. F. B. Mohamed Salleh and K. H. Low, "Concept of Operations (ConOps) for Traffic Management of Unmanned Aircraft Systems (TM-UAS) in Urban Environment," in *AIAA Information Systems-AIAA Infotech @ Aerospace*, no. January.

Reston, Virginia: American Institute of Aeronautics and Astronautics, jan 2017, pp. 1–13.

[21] J. Krozel, M. Peters, K. D. Bilimoria, C. Lee, and J. S. Mitchell, "System Performance Characteristics of Centralized and Decentralized Air Traffic Separation Strategies," *Air Traffic Control Quarterly*, vol. 9, no. 4, pp. 311–332, oct 2001.

[22] M. Ballin, J. Hoekstra, D. Wing, and G. Lohr, "NASA Langley and NLR Research of Distributed Air/Ground Traffic Management," in *AIAA's Aircraft Technology, Integration, and Operations (ATIO) 2002 Technical Forum*. Reston, Virigina: American Institute of Aeronautics and Astronautics, oct 2002, p. 5826.

[23] J. Klooster, S. Torres, D. Earman, M. Castillo-Effen, R. Subbu, L. Kammer, D. Chan, and T. Tomlinson, "Trajectory synchronization and negotiation in Trajectory Based Operations," in *29th Digital Avionics Systems Conference*. IEEE, oct 2010, pp. 1.A.3–1–1.A.3–11.

[24] K. Wichman, L. Lindberg, L. Kilchert, and O. Bleeker, "Four-Dimensional Trajectory Based Air Traffic Management," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*. Reston, Virigina: American Institute of Aeronautics and Astronautics, aug 2004, p. 5413.

[25] D.-S. Jang, C. A. Ippolito, S. Sankararaman, and V. Stepanyan, "Concepts of Airspace Structures and System Analysis for UAS Traffic flows for Urban Areas," in *AIAA Information Systems-AIAA Infotech @ Aerospace*, no. January. Reston, Virginia: American Institute of Aeronautics and Astronautics, jan 2017, p. 0449.

[26] E. Sunil, J. Ellerbroek, J. Hoekstra, A. Vidosavljevic, M. Arntzen, F. Bussink, and D. Nieuwenhuisen, "Analysis of Airspace Structure and Capacity for Decentralized Separation Using Fast-Time Simulations," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 1, pp. 38–51, 2017.

[27] S. Bae, H.-S. Shin, C.-H. Lee, and A. Tsourdos, "A New Multiple Flights Routing and Scheduling Algorithm in Terminal Manoeuvring Area," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*. IEEE, sep 2018, pp. 1–9.

[28] S. Bae, H.-S. Shin, and A. Tsourdos, "A New Graph-Based Flight Planning Algorithm for Unmanned Aircraft System Traffic Management," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*. IEEE, sep 2018, pp. 1–9.

[29] S. Bortoff, "Path planning for UAVs," *Proceedings of the 2000 American Control Conference. ACC (IEEE Cat. No.00CH36334)*, no. June, pp. 364–368 vol.1, 2000.

[30] J. Enright, E. Frazzoli, K. Savla, and F. Bullo, "On Multiple UAV Routing with Stochastic Targets: Performance Bounds and Algorithms," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, no. August. Reston, Virigina: American Institute of Aeronautics and Astronautics, aug 2005, p. 5830.

[31] G. B. Lamont, J. N. Slear, and K. Melendez, "UAV swarm mission planning and routing using multi-objective evolutionary algorithms," *IEEE Symposium Computational Intelligence in Multicriteria Decision Making*, no. Mcdm, pp. 10–20, 2007.

[32] J. Hall and D. Anderson, "Reactive route selection from pre-calculated trajectories – application to micro-UAV path planning," *The Aeronautical Journal*, vol. 115, no. 1172, pp. 635–640, oct 2011.

[33] L. De Filippis, G. Guglieri, and F. Quagliotti, "Path Planning Strategies for UAVS in 3D Environments," *Journal of Intelligent & Robotic Systems*, vol. 65, no. 1-4, pp. 247–264, jan 2012.

[34] W. Liu, Z. Zheng, and K.-Y. Cai, "Bi-level programming based real-time path planning for unmanned aerial vehicles," *Knowledge-Based Systems*, vol. 44, pp. 34–47, may 2013.

[35] A. Altmann, M. Niendorf, M. Bednar, and R. Reichel, "Improved 3D Interpolation-Based Path Planning for a Fixed-Wing Unmanned Aircraft," *Journal of Intelligent & Robotic Systems*, vol. 76, no. 1, pp. 185–197, sep 2014.

[36] M. Radmanesh and M. Kumar, "Flight formation of UAVs in presence of moving obstacles using fast-dynamic mixed integer linear programming," *Aerospace Science and Technology*, vol. 50, pp. 149–160, mar 2016.

[37] S. G. Manyam, S. Rasmussen, D. W. Casbeer, K. Kalyanam, and S. Manickam, "Multi-UAV routing for persistent intelligence surveillance & reconnaissance missions," *2017 International Conference on Unmanned Aircraft Systems, ICUAS 2017*, pp. 573–580, 2017.

[38] K. Yu, A. K. Budhiraja, and P. Tokekar, "Algorithms for Routing of Unmanned Aerial Vehicles with Mobile Recharging Stations," *IEEE International Conference on Robotics and Automation 2018*, pp. 1–5, mar 2017.

[39] K. Dorling, J. Heinrichs, G. G. Messier, and S. Magierowski, "Vehicle Routing Problems for Drone Delivery," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 1, pp. 70–85, 2017.

[40] R. T. Wong, "Combinatorial Optimization: Algorithms and Complexity (Christos H. Papadimitriou and Kenneth Steiglitz)," *SIAM Review*, vol. 25, no. 3, pp. 424–425, jul 1983.

[41] J. Y. Yen, "Finding the K Shortest Loopless Paths in a Network," *Management Science*, vol. 17, no. 11, pp. 712–716, jul 1971.

[42] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, dec 1959.

References for Chapter A

[1] D. G. Hull, *Fundamentals of airplane flight mechanics*, 2007.

[2] J. Roskam and C.-T. E. Lan, *Airplane aerodynamics and performance*. DARcorporation, 1997.

[3] Eurocontrol, "User Manual for the Base of Aircraft Data (Bada) Revision 3.12," Eurocontrol, Tech. Rep. August, 2014.

[4] M. Fernando and S. Arnedo, *Commercial Aircraft Trajectory Planning based on Multiphase Mixed-Integer Optimal Control by*, 2013, no. April.

[5] R. Bill, "Fuel Conservation Strategies: Cost Index Explained," *Boeing Aero Magazine*, pp. 26–28, 2007.

[6] M. A. Patterson and A. V. Rao, "GPOPS-II," *ACM Transactions on Mathematical Software*, vol. 41, no. 1, pp. 1–37, oct 2014. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2684421.2558904

[7] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, mar 2006. [Online]. Available: http://link.springer.com/10.1007/s10107-004-0559-y