Optimization of Arrival Air Traffic in the Terminal Area and in the Extended Airspace

Ying Huo

▶ **To cite this version:**

Ying Huo. Optimization of Arrival Air Traffic in the Terminal Area and in the Extended Airspace. Optimization and Control [math.OC]. Université Paul Sabatier - Toulouse III, 2022. English. NNT : 2022TOU30008 . tel-03710452

HAL Id: tel-03710452

https://theses.hal.science/tel-03710452v1

Submitted on 30 Jun 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Université Fédérale
Toulouse Midi-Pyrénées

THÈSE

En vue de l'obtention du
DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE
Délivré par l'Université Toulouse 3 - Paul Sabatier

Présentée et soutenue par

Ying HUO

Le 3 février 2022

Optimisation du trafic aérien à l'arrivée dans la zone terminale et dans l'espace aérien étendu

Ecole doctorale : **EDMITT - Ecole Doctorale Mathématiques, Informatique et Télécommunications de Toulouse**

Spécialité : **Mathématiques et Applications**

Unité de recherche :
ENAC-LAB - Laboratoire de Recherche ENAC

Thèse dirigée par
Daniel DELAHAYE et Mohammed SBIHI

Jury

M. Weigang LI, Rapporteur
M. Manuel SOLER, Rapporteur
Mme Fulya AYBEK CETEK, Examinatrice
M. Pierre MARECHAL, Examinateur
M. Daniel DELAHAYE, Directeur de thèse
M. Marcel MONGEAU, Président

Optimization of Arrival Air Traffic in the Terminal Area and in the Extended Airspace

by

Ying HUO

A thesis presented for the degree of
Doctor of Philosophy

Supervisors:

Daniel DELAHAYE
Mohammed SBIHI

Toulouse III - Paul Sabatier University
Toulouse doctoral school of Mathematics, computer science and telecommunication
ENAC-LAB

3rd February 2022

Résumé

Selon les prévisions à long terme du trafic aérien de l'Organisation de l'Aviation Civile Internationale (OACI) en 2018, le trafic mondial de passagers devrait augmenter de 4,2% par an de 2018 à 2038. Bien que l'épidémie de COVID-19 ait eu un impact énorme sur le transport aérien, il se rétablit progressivement. Dès lors, l'efficacité et la sécurité resteront les principales problématiques du trafic aérien, notamment au niveau de la piste qui est le principal goulot d'étranglement du système. Dans le domaine de la gestion du trafic aérien, la zone de manœuvre terminale (TMA) est l'une des zones les plus complexes à gérer. En conséquence, le développement d'outils d'aide à la décision pour gérer l'arrivée des avions est primordial. Dans cette thèse, nous proposons deux approaches d'optimisation qui visent à fournir des solutions de contrôle pour la gestion des arrivées dans la TMA et dans un horizon étendu intégrant la phase en route.

Premièrement, nous abordons le problème d'ordonnancement des avions sous incertitude dans la TMA. La quantification et la propagation de l'incertitude le long des routes sont réalisées grâce à un modèle de trajectoire qui représente les informations temporelles sous forme de variables aléatoires. La détection et la résolution des conflits sont effectuées à des points de cheminement d'un réseau prédéfini sur la base des informations temporelles prédites à partir de ce modèle. En minimisant l'espérance du nombre de conflits, les vols peuvent être bien séparés. Outre le modèle proposé, deux autres modèles de la litérrature - un modèle déterministe et un modèle intégrant des marges de séparation - sont présentés comme références. Un recuit simulé (SA) combiné à une fenêtre glissante temporelle est proposé pour résoudre une étude de cas de l'aéroport de Paris Charles de Gaulle (CDG). De plus, un cadre de simulation basé sur l'approche Monte-Carlo est implémenté pour perturber aléatoirement les horaires optimisés des trois modèles afin d'évaluer leurs performances. Les résultats statistiques montrent que le modèle proposé présente des avantages absolus dans l'absorption des conflits en cas d'incertitude.

Dans une deuxième partie, nous abordons un problème dynamique basé sur le concept de Gestion des Arrivées Étendue (E-AMAN). L'horizon E-AMAN est étendu jusqu'à 500 NM de l'aéroport de destination permettant ainsi une planification anticipée. Le caractère dynamique est traitée par la mise à jour périodique des informations de trajectoires réelles sur la base de l'approche par horizon glissant. Pour chaque horizon temporel, un sous-problème est établi avec pour objectif une somme pondérée de métriques de sécurité du segment en route et de la TMA. Une approche d'attribution dynamique des poids est proposée pour souligner le fait qu'à mesure qu'un aéronef se rapproche de la TMA, le poids de ses métriques associées à la TMA devrait augmenter. Une étude de cas est réalisée à partir des données réelles de l'aéroport de Paris CDG. Les résultats finaux montrent que grâce à cet ajustement anticipé, les heures d'arrivée des avions sont proches des heures

prévues tout en assurant la sécurité et en réduisant les attentes.

Dans la troisième partie de cette thèse, on propose un algorithme qui accélère le processus d'optimisation. Au lieu d'évaluer les performances de tous les aéronefs, les performances d'un seul aéronef sont concentrées dans la fonction objectif. Grâce à ce changement, le processus d'optimisation bénéficie d'une évaluation d'objectif rapide et d'une vitesse de convergence élevée. Afin de vérifier l'algorithme proposé, les résultats sont analysés en termes de temps d'exécution et de qualité des résultats par rapport à l'algorithme utilisé à l'origine.

Abstract

According to the long term air traffic forecasts done by International Civil Aviation Organization (ICAO) in 2018, global passenger traffic is expected to grow by 4.2% annually from 2018 to 2038 using the traffic data of 2018 as a baseline. Even though the outbreak of COVID-19 has caused a huge impact on the air transportation, it is gradually restoring. Considering the potential demand in future, air traffic efficiency and safety will remain critical issues to be considered. In the airspace system, the runway is the main bottleneck in the aviation chain. Moreover, in the domain of air traffic management, the Terminal Maneuvering Area (TMA) is one of the most complex areas with all arrivals converging to land. This motivates the development of suitable decision support tools for providing proper advisories for arrival management. In this thesis, we propose two optimization approaches that aim to provide suitable control solutions for arrival management in the TMA and in the extended horizon that includes the TMA and the enroute phase.

In the first part of this thesis, we address the aircraft scheduling problem under uncertainty in the TMA. Uncertainty quantification and propagation along the routes are realized in a trajectory model that formulates the time information as random variables. Conflict detection and resolution are performed at waypoints of a predefined network based on the predicted time information from the trajectory model. By minimizing the expected number of conflicts, consecutively operated flights can be well separated. Apart from the proposed model, two other models - the deterministic model and the model that incorporates separation buffers - are presented as benchmarks. Simulated annealing (SA) combined with the time decomposition sliding window approach is used for solving a case study of the Paris Charles de Gaulle (CDG) airport. Further, a simulation framework based on the Monte-Carlo approach is implemented to randomly perturb the optimized schedules of the three models so as to evaluate their performances. Statistical results show that the proposed model has absolute advantages in conflict absorption when uncertainty arises.

In the second part of this thesis, we address a dynamic/on-line problem based on the concept of Extended Arrival MANagement (E-AMAN). The E-AMAN horizon is extended up to 500NM from the destination airport so as to enhance the cooperation and situational awareness of the upstream sector control and the TMA control. The dynamic feature is addressed by periodically updating the real aircraft trajectory information based on the rolling horizon approach. For each time horizon, a sub-problem is established taking the weighted sum of safety metrics in the enroute segment and in the TMA as objective. A dynamic weights assignment approach is proposed to emphasize the fact that as an aircraft gets closer to the TMA, the weight for its metrics associated with the TMA should increase. A case study is carried out using the real arrival traffic data of the Paris CDG airport.

Final results show that through early adjustment, the arrival time of the aircraft can meet the required schedule for entering the TMA, thus ensuring overall safety and reducing holding time.

In the third part of this thesis, an algorithm that expedites the optimization process is proposed. Instead of evaluating the performance of all aircraft, single aircraft performance is focused and a corresponding objective function is created. Through this change, the optimization process benefits from fast evaluation of objective and high convergence speed. In order to verify the proposed algorithm, results are analyzed in terms of execution time and quality of result compared to the originally used algorithm.

Acknowledgment

First of all, I would like to express my sincere gratitude to my supervisor: Daniel Delahaye. From my internship until the finish of this thesis, he helped me to get on the research path, taught me to code, shared his brilliant ideas as well as his humour. My thanks equally goes to my co-supervisor: Mohammed Sbihi, who helped me with his massive mathematical knowledge and rigorous attitude. He encouraged me to express myself and gave me practical suggestions on the study. I received lots of intangible wealth from my two supervisors which would be beneficial for the rest of my life.

Many thanks to my jury members for being interested in my work and sharing their valuable time with me. Thanks to the reviewers, Weigang Li and Manuel Soler, for their detailed advice from the thesis structure to the algorithm that would help me improve this thesis. Thanks to the examiners: Fulya Aybek Cetek and Pierre Marechal for their insightful discussions regarding this thesis. Special thanks to the examiner Marcel Mongeau, who is a kind and considerate professor who showed a warm welcome from the beginning of my PhD study.

I feel so lucky that I have so many friends in ENAC to share my daily life. They brought me endless happiness and deep thinking about the meaning of life. I would like to take this opportunity to thank my friend Serge Roux for helping me with technical issues regarding software, computer systems and new tools, correcting my French and breaking my traditional impression of friendship. I will always miss you and bear in mind the happiness shared with you. Thanks to Ji Ma, who helped me get rid of the loneliness on the road of research. Thanks to my friends Shangrong Chen, Ruohao Zhang and Xiao Liang for the amazing adventures we have experienced together while discovering the world. Thanks to Alexandre Tran for teaching me to play tennis. Thanks to Sana Ikli for offering me so many words of comfort to ease my nerves. Thanks to Gilles Barion for inviting us to his place that full of surprises. Thanks to other friends who appeared in my life, you are all treasures to me: Zhengyi Wang, Thinh Hoang, Sylvian Roudiere, Hasan Heydari, Ziqing Li, Wenjing Liu, Paveen Juntama, Sashiko Shirai, Yeqing Li, Jiayi Chen, Geoffrey Scozzaro, Clara Buire. The memories of being with you will always stay with me and they make every moment of my PhD study vivid.

I would like to express my gratitude to the people I cherish the most: my family, who always support me unconditionally. Finally, a special thank you to my boyfriend Xiao HU, who plays the role of both friend and family, who accompanies me with all his patience and love, who gave me determination to persevere under difficulties and a bright hope for my life in the future. Just like what he said to me, I would like to say this back to him with all my heart: "Nothing is important without you".

Contents

List of Figures

List of Tables

Acronyms

ACC Area Control Center

ACO Ant Colony Optimization

ALP Aircraft Landing Problem

AMAN Arrival MANagement

ANSP Air Navigation Service Provider

ASM Airspace Management

ASP Aircraft Scheduling Problem

ASSP Aircraft Sequencing and Scheduling Problem

ATC Air Traffic Control

ATFM Air Traffic Flow Management

ATM Air Traffic Management

ATS Air Traffic Service

B&B Branch and Bound

CDG Charles de Gaulle

CDO Continuous Descent Operation

CPS Constrained Position Shifting

OSA Original Simulated Annealing

CTA Controlled Time of Arrival

DP Dynamic Programming

SSA Selective Simulated Annealing

E-AMAN Extended Arrival MANagement

ETA Estimated Time of Arrival

FAF Final Approach Fix

FCFS First-Come-First-Serve

FIR Flight Information Region

FMC Flight Management Computer

GA Genetic Algorithm

GDP Ground Delay Program

IAF Initial Approach Fix

ICAO International Civil Aviation Organization

IF Intermediate Fix

IFR Instrument Flight Rules

LP Linear Programming

MAP Missed Approach Point

MILP Mix Integer Linear Programming

MIP Mix Integer Programming

NextGen Next GENeration Air Transportation System

PBN Performance-Based Navigation

PMS Point Merge System

RTA Required Time of Arrival

RHC Receding Horizon Control

SA Simulated Annealing

SAA Sample Average Approach

SESAR Single European Sky ATM Research

SID Standard Instrument Departure Route

STAR Standard Terminal Arrival Route

SWIM System-Wide Information Management

TMA Terminal Maneuvering Area

TRACON Terminal Radar Approach CONtrol

TTG Time To Gain

TTL Time To Lose

TS Tabu Search

TSP Traveling Salesman Problem

VFR Visual Flight Rules

Chapter 1

Introduction

Air traffic plays an important role in the worldwide connections, which facilitates the development in varied domains such as tourism, education, commercial cooperation, etc. The long-term forecasts for the air traffic before 2020 have concluded a great increase in demand. Despite the outbreak of COVID-19 has caused a huge reduction in air traffic demand, gradual restoration is enabled due to the vaccination and proper quarantine regulations. It is evident that in a long run, air traffic demand will be at a high pace of growth and the studies that aim at improving the operational efficiency of air traffic management while ensuring safety are always of great importance. One of the great challenges will be the congestion in the airspace around the airport. Particularly for arrival air traffic, for which congestion will cause multifaceted impacts such as delay, economic loss and extra workload of air traffic controllers and aircrew. Therefore, this thesis focuses on optimizing the controls of arrival aircraft operations to an airport while considering the practical issue of uncertainty. In this chapter we will give the background of air traffic and its management in order to situate the issues addressed in this thesis.

This chapter is organized as follows: Section 1.1 outlines the current situation and the forecasts of air traffic demand before the outbreak of COVID-19. Then, the air traffic management system and the related services are described in Section 1.2. Section 1.3 introduces the aviation system, different kinds of controlled airspace, and the corresponding services provided for flights are also presented. Further, the addressed problems of this thesis are presented in Section 1.4. Finally, the structure of this thesis is given in Section 1.5.

1.1 Current situation and forecast of air transport

The sudden outbreak of the pandemic COVID-19 has led to a dramatic loss in the aviation industry. By April 2020, the number of global flights had dropped by nearly 80% with international flights being affected the most (ICAO, 2021*a*). With the efforts from governments and international organizations such as International Civil Aviation Organization (ICAO), World Health Organization (WHO), the air transport industry has gradually recovered, starting from the inner-continental operations (EUROCONTROL, 2020*b*). Still, the impact of COVID-19 lasts for the whole year of 2020, which leads to an average decline of 60% in world passenger traffic compared to the level of 2019. Estimation for

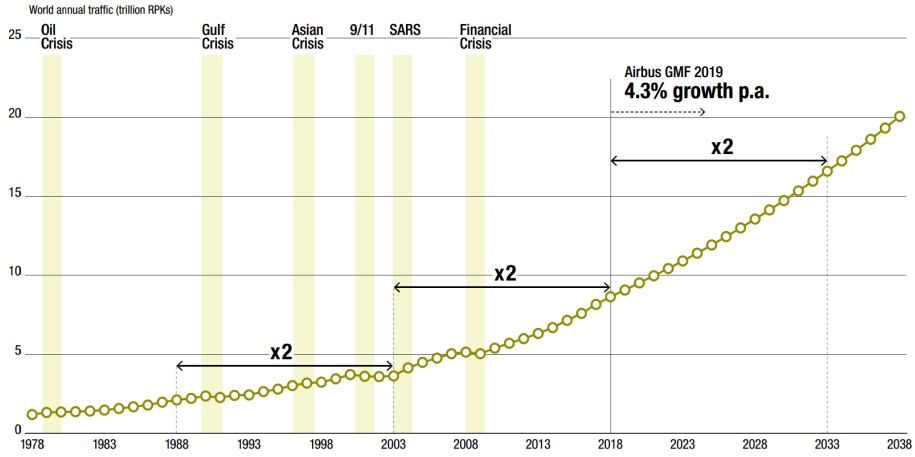


Figure 1.1: Total passenger traffic historical data and forecast shown in the Airbus global market forecast based on the source from ICAO before COVID-19.

2021 shows a reduction of 42% to 47% in passenger traffic (ICAO, 2021*a*). As air transport is not only a victim of COVID-19 epidemic, but also plays an instrumental role in enforcing the spread of the disease, the interaction between them makes the long-term impact of the epidemic unpredictable. However, it is evident that even though the epidemic will affect the aviation industry for several years and the predicted growth of air traffic will be realized with a delay (Gudmundsson *et al.*, 2021; EUROCONTROL, 2020*a*), air transportation will continuously restore with the aligned industry guidance and the medical treatment development. This optimistic expectation can be validated by some regional situations. For example, domestic air traffic in China exceeded the level of 2019 from the autumn of 2020 until January 2021 and again surpassed the level of 2019 in March (ICAO, 2021*a*).

When considering the potential growth of air traffic demand, forecasts conducted before the outbreak of COVID-19 can be a main reference. It predicted that annual passenger traffic growth is up to 4.2% from 2018 to 2038, and global freight traffic is expected to grow at 3.5% annually from 2015 to 2035 (ICAO, 2021*b*). Fig. 1.1 depicts the forecasts of the passenger demand based on historical data. Considering the challenges brought by the potential demand in the future, it has become a hot topic to strengthen air traffic management from the perspective of improving efficiency while maintaining safety.

1.2 Air traffic management

Air Traffic Management (ATM) is an operational conception requiring the total performance of all systems that assist aircraft in departing from an aerodrome, transiting airspace, and landing at a destination aerodrome to ensure safe, economical and efficient air traffic operations. ATM consists of three main services: Air Traffic Service (ATS), Air Traffic Flow Management (ATFM), Airspace Management (ASM). The services are performed in a collaborative manner while still having their own function domains.

1.2.1 Air Traffic Services (ATS)

ATS regulates and assists aircraft in real-time to ensure their safe operations. Various kinds of services are defined in the field of ATS such as air traffic advisory service, air traffic control service, area control service, approach control service, and aerodrome control service. ATS has multiple objectives to achieve and each of them is inherent to a particular service. Specifically, air traffic control service accomplishes the objectives of preventing all collisions between aircraft and expediting and maintaining an orderly flow of air traffic. Flight information service provides advice and information that is useful for Air Traffic Control (ATC). Alerting service aims to notify appropriate organizations regarding aircraft in need of search and rescue.

1.2.2 Air Traffic Flow Management (ATFM)

The objective of ATFM is to ensure optimal traffic flow when demand is expected to exceed the available capacity of the ATC system. ATC capacity reflects the ability of the system to provide service, and is expressed by the number of aircraft entering a specified portion of the airspace in a given time period. The causes of traffic congestion are various, such as conflicting with users' requirements, air navigation system limitations and unexpected weather conditions. In those cases, alleviating measures need to be considered. For example, holding, re-routing or deviations for the aircraft already in the air, and the ground delay program which holds the aircraft at the departure airport with engines off to save fuel and to reduce controller workload. In a dynamic environment, ATC is very time sensitive and should always be planned in advance. Specified in accordance to time, ATFM is done in three phases and explained as follows.

- **Strategic planning**

  This phase includes measures that are normally carried out well in advance from around one year down to one week before the real-time operations of the flight. This process aims to predict the air traffic load and make corresponding plans at a strategic level. The measures are results of coordination of multiple parties such as ATC service providers and aircraft operators. Planning belonging to this phase includes air route structure design, controller reassignment, flight schedule renewal.

- **Pre-tactical planning**

  This phase includes measures that are carried out six days to one day in advance. They can be seen as a fine tuning of the strategic measures using more-updated information about traffic, weather and facilities. This phase aims at optimizing the overall ATM network performance, minimizing delays and other costs by taking measures like re-routing, re-scheduling, etc. with respect to the decisions made in the strategic phase.

- **Tactical operations**

  In this phase, measures that take effect on the day of implementation are focused, including the execution of the previously agreed measures and the monitoring of the situation to ensure that the measures have desired effects. Detailed flight plans are

issued including aircraft departure slots, re-routings and alternative flight profiles in order to avoid congestion and enhance efficiency according to real-time traffic demand.

The ATFM strongly relies on the information, as in each phase, predictions of expected demand should be used for the formulation of the plans. Its coordination with the ATS authorities attempts to increase the available ATC capacity when there is a requirement. Even though ATC capacity shortfalls cannot be eliminated, the implementation of suitable tactical measures in the area of ATFM at a given time might mitigate the congestion. It is worth mentioning that in air traffic flow control, spacing of aircraft should not be mixed with separation, but rather should be based on an "acceptance rate", i.e. the number of aircraft accepted in a given time period. In addition, ATFM is not restricted in a single country, since the international air traffic has a high share among the total traffic volume.

1.2.3 Airspace Management (ASM)

The role of ASM is to manage the airspace in an efficient way. ASM is performed through the following functions: the design of airspace structures (airport control zone, terminal area, sectors), allocating airspace to its various users, dynamic management of the airspace structures in time and allocation of airspace between the different categories of users based on prescribed airspace use plan.

1.3 The complexity of aviation system

The dynamic and integrated air traffic operation requires the collaboration of all relevant parties to provide seamless services on the basis of facilities and airspace structures. The whole process is a long-distance management involving the airborne and ground-based functions, and the related facilities and resources are described as follows.

1.3.1 Ground system and services

Airports are complex sub-systems of the civil aviation system which handle passenger services, aircraft operations, commercial centers and connections to public transport, etc. From an aircraft operation perspective, the primary functions of an airport involve facilitating landing and takeoff of aircraft, ground maneuvering, the positioning and parking of aircraft. During the operation in the airport, an aircraft uses physical facilities including runways, taxiways, aprons, strips, terminals, etc. Each component carries out a unique function in order to finally load and unload passengers and cargo.

The complexities of a modern airport are not only reflected in the varied ground operations but also reflected in the connections between airborne and airside operations. Two facilities in the airport are especially significant for the transition. First is the runway. The choices of the locations and the orientation of the runways for an airport are mainly dependent on the prevalence of wind condition at this site. As the wind direction has a great impact on aircraft performance during landing and takeoff, a runway can normally be used in both directions. Usually the most suitable directions are used taking into account the types of aircraft and the wind direction. The other important facility is the

airport control tower. The control tower is a tall and window-structured building located on the airport ground to provide an open and clear view that facilitates the observation of aircraft. The specific function of the control tower is airfield (airport) control, which can fall into two or even three sections: local control (air control) and ground control (pre-flight for the third section). It can also carry out approach control in the terminal zone, or simply provide alerting service. The two main control services of the control tower are introduced as follows:

- **Air control:** It is responsible for the active runway surfaces. Air traffic controllers apply the procedures and runway separation rules to orderly and safely operate the air traffic in the vicinity of the airport and on the runways.
- **Ground control:** It is responsible for the aircraft operation on the taxiway, holding areas, some transitional aprons, gate parking.

Within the tower, highly disciplined communication processes between air control and ground control are applied. Any operations that are influential to the further operations need to be aware in advance. For example, upstream approach radar controllers need to cooperate with air control constraints to adjust the aircraft landing time considering both runway and taxiway availabilities. Ground control requires that the controllers are well aware of the aircraft heading to the runways so as to maximize runway utilisation through effective departure spacing. Fig. 1.2 illustrates the control process in the airport.

It is worth mentioning that most airports have been equipped with surveillance displays. Controllers can use a radar system that integrates geographical information and aircraft operational information to identify the overall traffic situation. Then, the communications are normally done via radio.

1.3.2 Airspace configuration

The airspace for aircraft operations around the world is divided into Flight Information Region (FIR)s. Each FIR is managed by a control authority that has responsibility to ensure that air traffic services are provided to the aircraft flying within it. It is possible to define portions of controlled airspace within a FIR. The structures are called control zones (CTZs) if their lower limit is the ground and others are called control areas (CTAs). FIR is divided into six classes, based on which air traffic services are provided for aircraft with different flight rules (Visual Flight Rules (VFR) or Instrument Flight Rules (IFR)), where classes A, B, C, D and E are areas of controlled airspace and G is uncontrolled airspace (ICAO, 2017). In the airspace, two kinds of controls are performed by the controllers in the field of ATS.

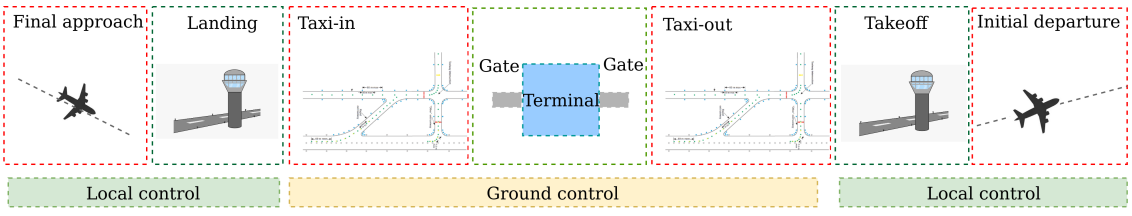


Figure 1.2: Airport control processes.

- **Terminal control:** In the terminal area (control area), terminal control is carried out based on the radar control facility. In the US, it is referred to as Terminal Radar Approach CONtrol (TRACON). Terminal controllers are responsible for providing all ATC services in a 30 to 50 nautical mile (56 to 93 km) radius from the airport. Traffic flow associated with this area is broadly divided into departures, arrivals, and overflights. The arrival traffic is handled by the sector controller to the terminal control with suitable rates and altitudes. When an aircraft enters the final approach, it is then transferred to the airport local control. The departure aircraft are handed off to terminal control after takeoff, and then follow the departure route until they are further transferred to the next control segment. Terminal control is responsible for ensuring that aircraft are at an appropriate altitude when they are handed off, and that aircraft arrive at a suitable rate for landing.

- **Enroute control:** In the enroute segment, aircraft are controlled by the enroute air traffic controllers who work in facilities called air traffic control centers or Area Control Center (ACC). Center controllers are responsible for issuing instructions to pilots about climbing or descending their aircraft to the assigned altitudes while ensuring proper separation of an aircraft from all other aircraft in the immediate area. Additionally, the aircraft must be placed in a flow consistent with the aircraft's route from the flight plan. This effort is complicated by crossing traffic, severe weather, special missions that require large airspace allocations and traffic density. When an aircraft reaches the boundary of a center's control area, it is handed off to the next ACC or to the terminal control.

Different control services are provided for different flight phases. Fig. 1.3 illustrates the aircraft operation phases and the associated controlled airspace, where ground control and air control for the airport and its vicinity is described earlier. For the large airspace area, center controllers typically use long range radar to see aircraft within 200 nautical miles of the radar antenna. They may also use radar data to control when it provides a better vision of the traffic. The ocean areas are FIRs without radar systems available, so oceanic controllers provide ATC services using procedural control. These procedures use aircraft position reports, time, altitude, distance, and speed to ensure separation. Controllers record information on flight progress strips and input the flight positions reported by

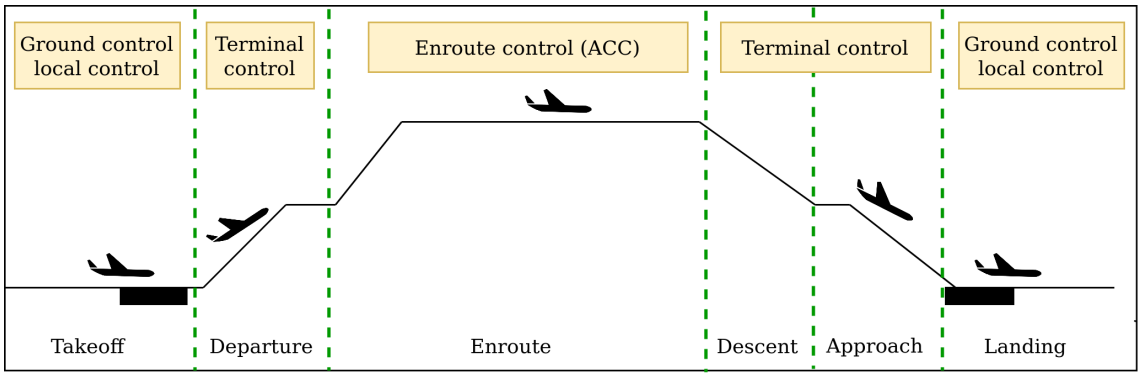


Figure 1.3: Fight phases and corresponding control services.

pilots in specially developed oceanic computer systems. The loss of accuracy of position track in this process requires greater separation distances for aircraft, which reduces the overall capacity for any given route.

When aircraft operate on the route, ATC and pilots need to make sure that the aircraft will remain on the route with the required parameters. The Flight Management Computer (FMC) is a computer system that uses a database to allow routes to be pre-programmed. The system is constantly updated with aircraft position by referring to available navigation aids (including VOR, DME, GNSS, IRU or a combination of these). A high-end FMC is capable of four-dimensional area navigation (latitude, longitude, altitude, time), and it can optimize aircraft performance to achieve the most possible economical flight. It also allows the pilots to input the whole flight plan and to modify it in flight (EUROCONTROL, 2013). All FMC contain a navigation database. The navigation database contains the elements from which the flight plan is constructed. Modern FMC is not equipped on all flights, therefore, the performance of flight to accurately follow the instructions of controller varies.

1.3.3 ATM support systems

Nowadays, two major ATM projects , the Next GENeration Air Transportation System (NextGen) and Single European Sky ATM Research (SESAR) are carried out in the U.S. and in Europe to contribute to ATM modernization based on their specific situations. One common goal of these two projects is the development and implementation of the new technologies of communication, navigation and surveillance for the ATM systems (SESAR-FAA, 2017).

- **Communication system:** This system enables interactions between pilots and air traffic controllers. Currently, there are several types of communications between traffic controllers and pilots. The two most used methods are: multi-mode Very High Frequency (VHF)/ Ultra High Frequency(UHF) ground-air-ground communications based on radio stations, and ground-air-ground datalink support for air traffic management, in which datalink is a digital alternative to supplement the congested communication bands used in aviation.

- **Navigation system:** This system guarantees the aircraft to follow the pre-defined route and to access airports safely. Different navigation methods are used depending on whether the aircraft is flying under VFR or IFR. For VFR, pilots can choose to navigate using "dead reckoning" combined with visual observations with reference to appropriate maps. For IFR, instruments and radio navigation aids such as beacons can be used. Moreover, pilots can also be directed under radar control by air traffic controllers. The Performance-Based Navigation (PBN) (ICAO, 2008), including Area navigation (RNAV) and Required Navigation Performance (RNP), is one of the key concepts to enable more precise navigation in the modernized ATM system.

- **Surveillance system:** This system is used to determine the positions of aircraft and improve the situation awareness to prevent potential safety issues. The main applications of surveillance are for area control and approach control. The most common sensors used now are: Primary Surveillance Radar (PSR) and Secondary

Surveillance Radar (SSR), which rely on the signal receive-and-reply time to locate the aircraft. Additionally, Automatic Dependent Surveillance Broadcast (ADS-B) is implemented, where aircraft computes its position using its navigation systems (e.g. GNSS, inertial navigation system (INS), etc.) and transmits the position information at regular (short) intervals. The signal is received and decoded by ground stations and the aircraft position is extracted.

Communications, navigation and surveillance (CNS) infrastructures, and the radio spectrum they require, are the foundation of aviation operational performance, enabling airspace capacity. Without them, modern air transport would not exist. Through enhancing the resilience of the CNS infrastructure, safe, secure, predictable, efficient and sustainable operations can be delivered such that the performance of air network can be improved.

1.3.4 Aircraft arrival procedures

In this thesis, we specifically focus on the arrival air traffic. The arrival air traffic are merged in the controlled airspace surrounding the airport, which is the Terminal Maneuvering Area (TMA). This controlled area serves as a transitional region between enroute and airfield where there has a high volume of traffic. Terminal area is usually designed to a circular configuration centered on the geographic coordinates of either a single or set of airports. The scale of the TMA is usually defined in accordance with the traffic density that needs to be handled, and airspace boundaries of the TMA vary widely from airport to airport. Fig. 1.4 shows an example of the TMA of the Paris region.

In the bottom of the figure, a vertical view shows the altitudes of corresponding zones. Around the boundary of the TMA, enter (resp. exit) fixes are located, from which air traffic gets in (resp. gets out) of the TMA. Both the arrival traffic and the departure traffic are operated based on the predefined procedures and routes. The route sections connecting the TMA entry fixes to the Initial Approach Fix (IAF) is called Standard Terminal Arrival Route (STAR) and Standard Instrument Departure Route (SID) for arrival and departure, respectively. After getting through the STAR, aircraft can arrive at IAF to proceed three approach segments:

- **Initial approach segment**: This segment begins at an IAF and ends at the intermediate approach segment or Intermediate Fix (IF). In this segment, descent and procedural turn are performed to align the aircraft with the intermediate or final approach segment. There is a holding pattern located at the IAF.

- **Intermediate approach segment**: This segment positions the aircraft for the final descent to the airport. It begins at the IF and ends at the Final Approach Fix (FAF).

- **Final approach segment**: This segment begins at the FAF and ends at the Missed Approach Point (MAP). The alignment and descent for landing are accomplished in this segment.

- **Missed approach segment**: This segment starts at the MAP. It is designed to provide protection from obstacles throughout the Missed Approach manoeuvre. It

specifies a point where the missed approach begins, and a point or an altitude/height where it ends.

Fig. 1.5 illustrates the approach segments that constitute the overall approach of an aircraft. One additional segment is represented which is the arrival segment. This segment is a transition from the en-route phase to the approach phase of an aircraft. In Fig. 1.6, the real ATS in the surrounding airspace of the Paris Charles de Gaulle (CDG) airport is displayed. These three figures correspond to the above-mentioned procedures, including STAR, arrival route and final approach. In Fig. 1.6, the red boxes indicate the critical points that specify different approach segments.

When an aircraft is under radar control, the air traffic controller uses radar vectoring to perform navigational guidance for this aircraft. The traffic management in the TMA is subject to numerous constraints with a high traffic density and limited airspace capacity. During the operation, aircraft separation is one of the most important constraints that requires a high level of compliance. In the TMA, the standard separation norm between aircraft in the TMA is 3NM in the horizontal plane or 1000ft in the vertical plane as shown in Fig. 1.7. In addition to traffic separation standards applied to aircraft in the approach phase, more restrictive separation minima of wake vortex turbulence separation requirements are defined to ensure that the following aircraft, especially smaller ones, are

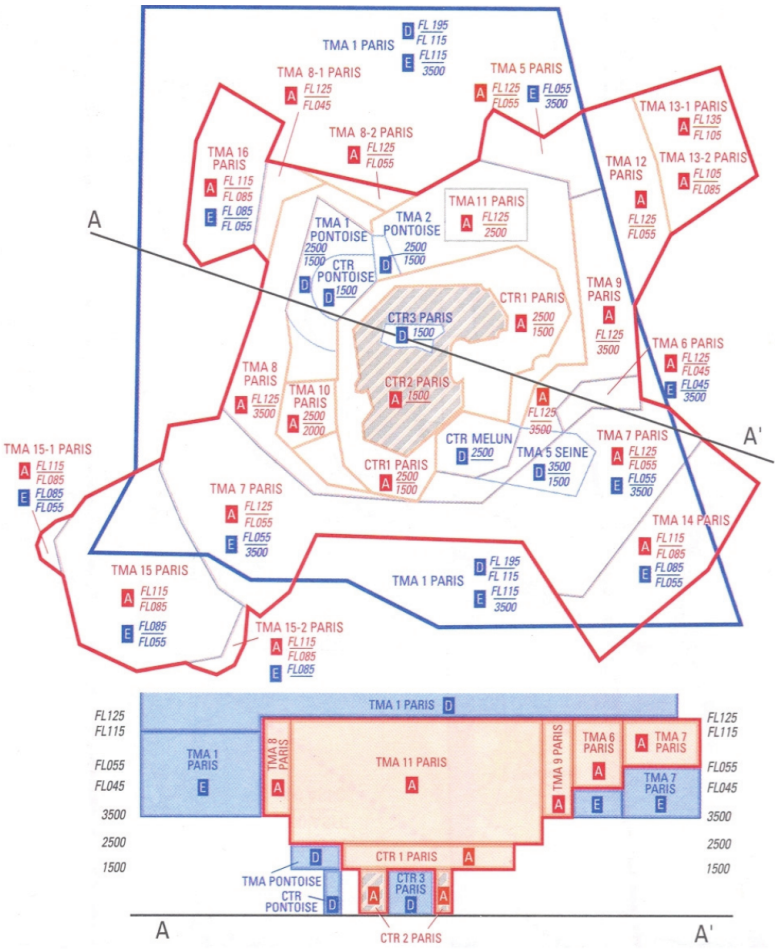


Figure 1.4: TMA of the Paris region.

not hazarded by the effects of wake vortex turbulence generated by a preceding aircraft. Wake vortex turbulence is defined as turbulence which is generated by the passage of an aircraft in flight. It will be generated from the point when the nose landing gear of an aircraft leaves the ground on take off and will cease to be generated when the nose landing gear touches the ground during landing. The origin of counter-rotating wing tip vortex is a direct and automatic consequence of the generation of lift by a wing. The strength of the vortex is governed by the weight, speed, and shape of the wing of the generating aircraft. The potential for hazardous wake vortex turbulence is greatest where aircraft follow the same tracks. This situation is mostly encountered close to the ground in the vicinity of airports where aircraft are in approach phase or departure phase of particular runways at high frequencies. Vortex typically persists for between one and three minutes. ATC provides standard separation requirements for all departing aircraft and for IFR traffic on approach. Separation requirements between aircraft depend on the associated size of the aircraft. Fig. 1.8 displays the impact of the wake vortex turbulence on the following aircraft by showing the generation of the turbulence and the related air movements.

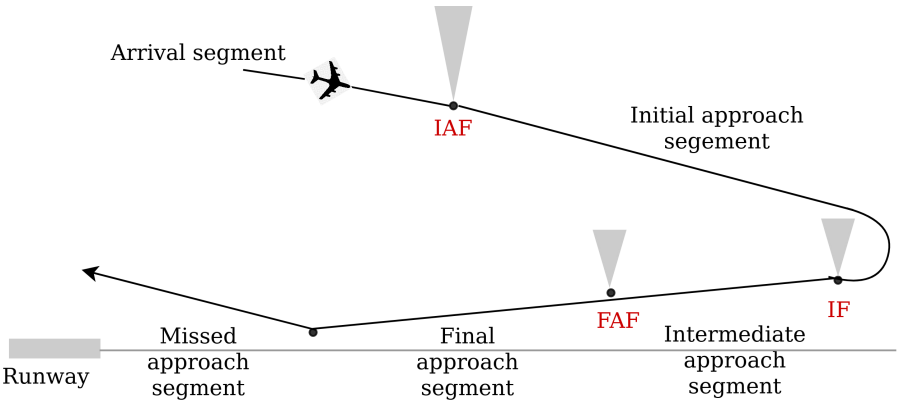
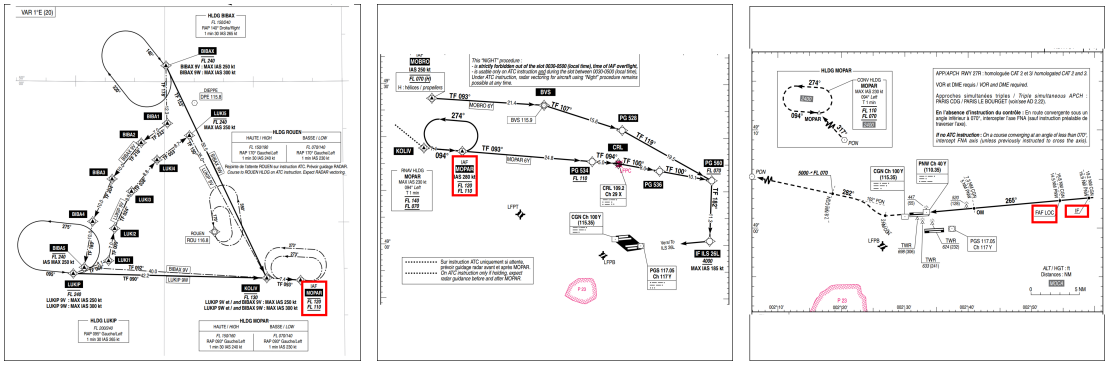


Figure 1.5: Approach phase is divided into three respective segments: the initial approach segment which starts from the IAF, the intermediate approach segment that starts from IF, and the final approach segment that starts from the FAF.



(a) STAR (b) Arrival route (c) Final approach

Figure 1.6: The STAR and the following route of MOPAR (IAF) associated with the Paris CDG airport. Red boxes indicate the critical points for entering different approach segments.

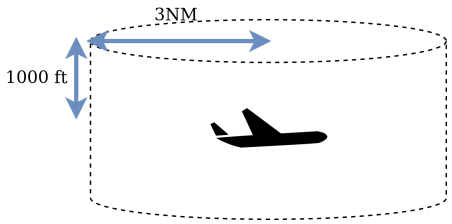


Figure 1.7: Standards separation norm in the TMA.

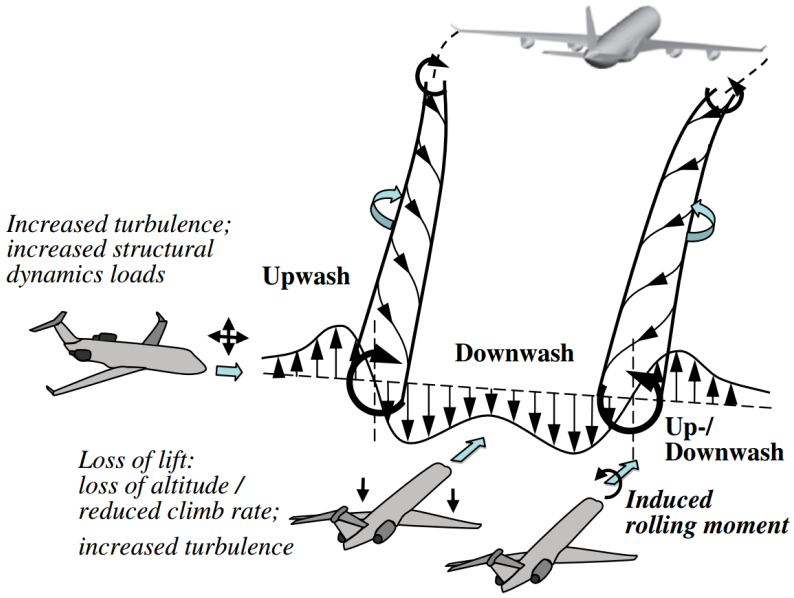


Figure 1.8: The impact of wake vortex turbulence on the following aircraft (Breitsamter, 2011).

1.4 Problems addressed in this thesis

All the prescribed context provides basic knowledge of flight operations and control systems. Based on all the information, challenges arise with the future growth of air traffic demand. Airports, especially the hub-airports will suffer from congestion and delays, which will cause safety risks, financial and environmental inefficiency.

Regarding the flight operations in airports or around airports. Arrival, departure and surface operations are managed by using the associated decision support tools for air traffic controllers to establish efficient aircraft sequence and schedule while considering the associated separation requirements. The task of managing flights is complex when high traffic volume is involved. Compared to departure flights and flights on the ground, arrival flights naturally experience more uncertainty, high delay costs, and higher workload for air traffic controllers and pilots. Therefore, it is necessary to develop optimization techniques for improving the operational efficiency of arrival air traffic while considering the uncertainty which is a critical factor causing safety problems and efficiency reduction.

The current arrival management relies on the development and deployment of the decision support tool named Arrival Manager, which is designed to provide automated

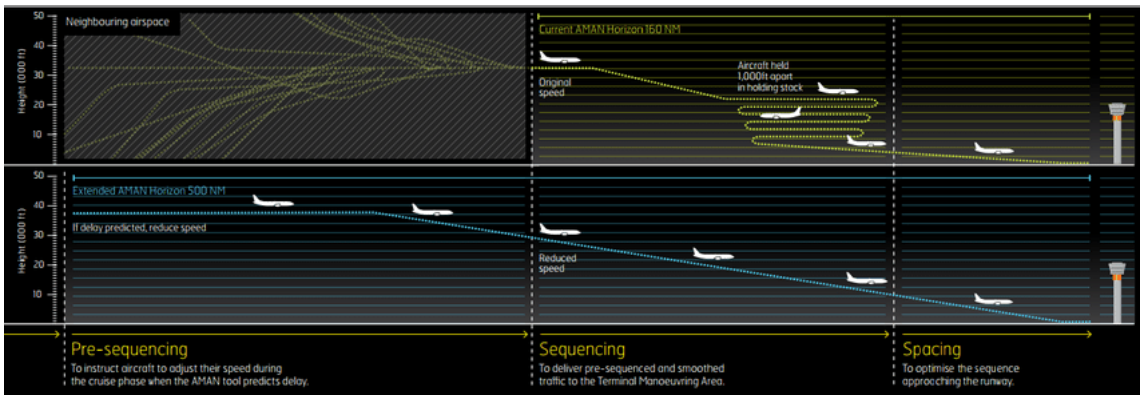


Figure 1.9: E-AMAN schematic compared to the original operational strategy (Peters, 2017).

sequencing support for the air traffic controllers handling air traffic that arrives at an airport. The current arrival management focuses on aircraft operations in the TMA and it aims to assist in reducing low-level holding and tactical intervention. However, based on this system, holding and stacking are still commonly used for aircraft to wait for proper landing slots, which will unavoidably lead to extra fuel consumption, noise and pollution. This is where the new concept of Extended Arrival MANagement (E-AMAN) is put forward by SESAR. This concept aims to enhance the cooperation between the ACC and the terminal control by implementing an earlier enroute speed control and sequencing so as to reduce the need of holding and stacking over the destination airport, and therefore achieving benefits such as efficiency, fuel saving and emission reduction etc. A comparison between the E-AMAN concept and the currently applied Arrival MANagement (AMAN) is shown in Fig. 1.9, from which different control strategies are illustrated. In this thesis, problems related to arrival air traffic are addressed from two ranges. First, a problem regarding arrival aircraft operations in the TMA is focused. Then the second study accommodates the E-AMAN concept and considers the TMA and its upstream enroute airspace. Due to the differences in management range, uncertainty need to be considered specifically for each problem. The addressed problems and the associated contributions are stated as follows:

- **Arrival air traffic optimization in the terminal area under uncertainty**
  The first problem that we addressed in this thesis is an arrival aircraft scheduling problem under uncertainty. We proposed a novel method to generate a robust arrival schedule that can potentially improve the ability of hedging against uncertainties during flight operations while still satisfying the various constraints required for safety in the TMA. In this problem, uncertainty is formulated directly by representing the time information of an aircraft arriving at a specific point by random variables. Uncertainty quantification and propagation along the routes are reflected in the trajectory model. For each aircraft, the random variables that represent time information for adjacent waypoints follow the Markov assumption. Based on the time information, conflict detection is performed in the predefined network. By taking advantage of the characteristics of normal distribution, an analytical evaluation of interactions between aircraft in terms of conflict measurement is obtained. The

objective function is then set to be the expected number of conflicts occurring in the network, which enables adjustment of the separations between aircraft pairs considering the available temporal and spatial resources. In our study, uncertainty is considered as an integrated element. This increases the adaptability of this model to the real-world application. Except for the proposed model, two other models: a deterministic model that does not consider uncertainty and a model that incorporates separation buffers, are presented as benchmarks. A meta-heuristic simulated annealing algorithm combined with the time decomposition sliding window is used as solution algorithm. Further, a simulation framework based on the Mont-Carlo approach is implemented for evaluating the optimized solutions obtained from the three models. Statistical comparison shows that the proposed model has absolute advantages with respect to the other two models in conflict absorbing ability when uncertainty arises. The complete research has been published (Huo *et al.*, 2021).

- **Arrival air traffic optimization in the TMA and the extended horizon**
  The second problem addresses the long-term arrival management following the concept of E-AMAN. To deal with this problem, we need to pay attention to three practical concerns. First, it is mentioned that one characteristic of this concept is to extend the arrival management range to enroute phase, with which, varied uncertainties may appear during flight operations. Second, due to different control manners, objectives and decisions of the enroute phase and the TMA are different. Third, as a flight gets closer to the TMA, its operational status reflected on objectives gets more important in the TMA. Based on these concerns, we choose to establish a dynamic/on-line control problem, where the aircraft operation process is updated regularly taking the updated decisions into account. Situations in the enroute segment and in the terminal area are considered separately and evaluated with different metrics considering different control standards. In the enroute segment, air traffic flow capacities are measured based on the radar separation minimum. Moreover, for flights that converge to the same waypoint, traffic merging workload is evaluated. While in the terminal area, conflict detection approaches which are identical to the deterministic model introduced previously (namely, evaluating the conflicts between two successive aircraft on sepecific waypoints) are applied to finely organize all the arrivals in the TMA. The objective is to reduce the route congestion, the overload of conflict resolution workload for enroute air traffic merging, and the number of conflicts in the TMA. A dynamic weights assignment approach is applied for each aircraft, where the weights assigned for the two segments depend on their locations. More precisely, an aircraft that is far away from the terminal area is assigned with a relative high weight for the enroute congestion and exceeded workload, whereas an aircraft that is near the terminal area is assigned with a high weight for the conflicts in the TMA. The decision change for each aircraft is then a result of balancing the congestion and workload in the enroute segment and the conflicts in the TMA taking into account the priority of the problems that need to be solved. Through this design, efficiency can be achieved in the aspects such as delay absorption, workload reduction, and fuel saving. This work has been submitted to a journal.

- **An improved algorithm for arrival aircraft scheduling problem in the TMA**
  Apart from the previously mentioned problems. We also address a problem which aims at improving computational efficiency. Compared to the models established in the previous two problems, instead of considering the conflicts of all flights in the objective function, a new algorithm that focuses on the performance of a single aircraft and minimizes the maximum individual cost among all flights is proposed. The cost here refers to the conflict related metric. The Simulated Annealing (SA) algorithm is adopted as solution algorithm, and the optimization details are demonstrated. For the proposed algorithm, by focusing on individual performance, the optimization process can be effectively guided by specifying flights with the worse performances. Moreover, information aggregation of all flights is not necessary, therefore efficiency and flexibility of the proposed algorithm are enhanced. The performances are analyzed in terms of the execution time and result quality based on a case study at the Paris CDG airport. Final results show an absolute advantage of the proposed algorithm in terms of computational time. This research work has been presented at the conference of SESAR Innovation Days in 2020.

1.5 Thesis structure

The plan of this thesis is as follows: Chapter 2 reviews existing studies in the literature that are related to the problems focused in this thesis. Chapter 3 addresses the arrival aircraft scheduling optimization problem in the TMA under uncertainty. In Chapter 4, we describe the details of the concept of E-AMAN, based on which a dynamic/on-line problem is formulated by considering different objectives and decisions of flights in the enroute phase and in the TMA. In Chapter 5, the single aircraft performance based algorithm for arrival aircraft scheduling problem is presented. SA is adapted accordingly to solve the problem. Finally, in Chapter 6, we conclude the whole thesis.

Chapter 2

Literature review

In this chapter, the research works related to the problems addressed in this thesis are discussed, which are the arrival air traffic optimization problems in different ranges and air traffic optimization problems under uncertainty. Specifically, Section 2.1 provides a general introduction of the research direction of this thesis. Section 2.2 investigates the aircraft scheduling problem at the runway and in the terminal area. Section 2.3 gives a review about the air traffic optimization problem under uncertainty. In Section 2.4, the concept of Extended Arrival MANagement (E-AMAN) is briefly introduced, and research works focusing on an extended horizon are discussed. For problem solving, Section 2.5 summarizes the popular solution algorithms including the exact, heuristic and meta-heuristic algorithms. Section 2.6 introduces the Simulated Annealing (SA) algorithm which is used to solve the problems that are addressed in this thesis. Finally Section 2.7 concludes this chapter.

2.1 Introduction

To ensure safety and efficiency, air traffic management is supported by complex systems that continuously provide control information for air traffic controllers and pilots based on various requirements. Arrival MANagement (AMAN), which is very important for the operations near the airport, has been widely concerned by researchers. The control support systems associated with the AMAN assist air traffic controllers in facilitating efficient arrival sequences and aircraft schedules to ensure smooth operation. The realization of this process leads to the optimization of Aircraft Sequencing and Scheduling Problem (ASSP). In accordance with geographical features, the scheduling problems are classified to *basic* and *detailed* (Samà *et al.*, 2013), which referring to the models incorporating only runways, and the models that consider other relevant resources in the Terminal Maneuvering Area (TMA), respectively.

The aircraft runway scheduling problem that focuses on a single resource is easier to be modeled with its typical features, and thus facilitates the aggregation of different interests in the models. The aircraft runway scheduling problem is generally split into two types of problems, involving either landings or take-offs, in which the arrival operation referred to as Aircraft Landing Problem (ALP) is discussed. Considering the route structure and the function of the TMA, the arrival operations in this area are subject to complicated

constraints and attract attention from the researchers. Since the configuration of TMA is uniquely associated with its airport, the studies that focus on a larger area of terminal airspace and runway provide greater level of detail in the model formulation. The above-mentioned problems have been reviewed in some works (Messaoud, 2020; Ikli *et al.*, 2021; Bennell *et al.*, 2011). In reality, air traffic control are carried out in various situations due to time and geographical locations change, so there are many research directions related to AMAN, among them, the works closely related to our study directions are presented and discussed.

2.2 Sub-problems arising in aircraft arrival management

2.2.1 Aircraft landing problem (ALP)

As the main bottleneck in the air traffic management system, runway operational efficiency is significant and needs to be improved. Therefore, aircraft landing and takeoff problems have been targeted intensively in decades. There exist several configurations of runway alignments that are established based on the airport environments such as altitude, prevailing wind directions, and desired capacities of the airports. Hence, for different airports, constraints that need to be satisfied are differentiated. While basically, some operational constraints are commonly applied to aircraft sequencing and scheduling problems. These constraints are:

- **Minimum separation constraints:** The separation requirements are hard constraints in the aircraft operation. The wake-vortex generated by the preceding aircraft affects the stability of the aircraft landing after it. The effect fades as the separation between two aircraft increases. The minimum wake turbulence separation requirements are therefore specified for the successively operated aircraft according to their associated categories defined by the take-off weight capacity. Moreover, the radar separation for radar control is also applied. Under normal circumstances, this value is set to 3 NM in the terminal area, details regarding this standard can be found in ICAO (2016).

- **Precedence constraints:** The precedence constraints refer to the unchangeable sequence or limited number of overtaking among aircraft due to the route structures, aircraft performance limitations, airline equity, etc., belonging to which, the first commonly applied condition is the First-Come-First-Serve (FCFS) principle (Beasley *et al.*, 2000; Soomer and Franx, 2008). The FCFS approach ensures a rather fair environment for aircraft by respecting their initial arrival sequences. Moreover, an explicit sequence is easy to implement and the workload of traffic controllers can be reduced. However, the strict implementation of FCFS can lead to a reduction of runway throughput. In order to improve runway utilization, another practical consideration is known as Constrained Position Shifting (CPS), which is firstly introduced to the air traffic optimization problem by Dear (1976). The CPS enhances the efficiency of the aircraft landing on the runway by filling the time vacancies with aircraft that are supposed to land later according to the FCFS approach. The limited allowable number of position shifting (overtaking) between aircraft avoids the

loss of equity while improving runway throughput to a great extent. The comparison of the two strategies is conducted in plenty of works showing that FCFS limits the performance of the solutions (Balakrishnan and Chandran, 2006, 2010).

- **Time window constraints:** Regarding the real operation of fights, aircraft landing time is bounded by the earliest landing time and the latest landing time. Those times are determined based on the available speed and maximum holding time, etc. The time window constraint regarding the aircraft arrival time was presented in Beasley *et al.* (2000) and then it is broadly used in other works.

The air transportation system is complex and involves many stakeholders, and each of them aims to achieve different interests that meet their needs. The common interests are safety and efficiency. While from some practical concerns, airlines care about the equity and fuel efficiency, and the governments pay more attention to environmental effects such as noise and emission. Based on all these interests, the commonly addressed objectives are classified as follows:

- **Runway throughput or makespan:** These are objectives used to maximize the runway capacity. Throughput refers to the number of aircraft operations that can be accommodated on a runway in a fixed time period. Makespan refers to the shortest time used to successfully operate a certain number of aircraft for landing. These two objectives evaluate the runway operation efficiency from two perspectives.

- **Delay or time deviation:** Delay can be referred to as the extra time spent after the required schedule time. In most cases, both early and late arrivals are not preferred for an aircraft. Therefore, some works use the time deviations between the scheduled landing time and the real landing time in the objective function. This is also referred to as a punctuality objective.

- **Environment costs:** The major environmental issues are fuel consumption, carbon emission, and the noise footprint on the ground.

- **Equity:** This is related to the landing sequence of flights. Airlines are especially sensitive to this parameter. FCFS policy can ensure the equity to some extent.

The ALP is usually established based on the above-mentioned constraints and objectives. The objectives chosen for a study rely on their specific interests, and both single objective function and multi-objective function are applied in the prior literature.

- **Single objective optimization for ALP**:
  Minimizing the total cost of delay is the most common objective. In Beasley *et al.* (2000), a cost function composed of linear cost relations of the positive and negative time deviations with respect to the target landing time is applied and a Mix Integer Programming (MIP) formulation is presented for both the single runway case and the multiple runway case. In this problem, time window constraint, separation constraint and workload constraint are incorporated. This is probably the most cited work, based on which, some works reformulated or tightened the constraints in accordance with their specific problems (Furini *et al.*, 2015; Farhadi *et al.*, 2014;

Lieder *et al.*, 2015; Prakash *et al.*, 2018). Many other works have implemented similar cost functions for time deviations with different relations. These functions can be either linear, piece-wise linear, or non-linear depending on the preference of delay and earliness (Girish, 2016; Salehipour, 2020; Vadlamani and Hosseini, 2014; Lieder *et al.*, 2015). Balakrishnan and Chandran (2006, 2010) addressed a single runway optimization problem which tried to maximize the runway throughput while incorporating the constraints of CPS. The model was established to accommodate the Dynamic Programming (DP). Psaraftis (1978) investigated two objectives separately, the first one tried to minimize the runway throughput, while the other one minimized the total passenger waiting time with two versions of ALP: single runway case with CPS, and two runways case without constraints for position shifting. In this study, DP is applied to solve the ALP. Briskorn and Stolletz (2014) formulated the ALP as a machine-scheduling model by representing runways as machines and aircraft as jobs to minimize the overall delay cost, in which aircraft were specified with classes for defining the job length.

- **Multi-objective optimization for ALP:**
  Multi-objective models put together and optimize the controversial objectives simultaneously to achieve a balance or discover the trade-offs of the concerned goals. Based on the model structure proposed in Balakrishnan and Chandran (2006), Lee and Balakrishnan (2008) analyzed the trade-offs between two sets of objective combinations, which are the trade-off between throughput and fuel costs, and the trade-off between throughput and operating costs. Soomer and Franx (2008) addressed the airlines equity by establishing specific cost functions for delay time in accordance with the particular interests of airlines. This study implemented a local search heuristic algorithm to determine the sequence of aircraft, based on which the Mix Integer Linear Programming (MILP) model is transferred to a Linear Programming (LP) to evaluate the performance of the result. Mokhtarimousavi *et al.* (2015) has proposed a three-objective model to minimize the total landing time (i.e. maximizing the runway throughput), the fuel consumption, and the costs of extra time staying in the apron and parking slot since the airport operation is considered. In addition, Bennell *et al.* (2017) proposed a multi-objective formulation taking into account runway throughput, time deviation from target time, and the fuel cost. Zhang *et al.* (2020) elaborated the inner relations between the commonly used criteria: min-max, min-sum, completion time related, and due-dates related criteria for the objective functions of ALP. Several appropriate objectives are chosen based on a theoretical analysis to establish a multi-objective model for the single runway case. The objectives are dedicated to achieving a smooth and efficient aircraft landing process with varied interpretations.

The above-mentioned studies mainly consider the ALP as static cases. However, dynamic cases that focus on real-time operations are also studied by lots of researchers. Beasley *et al.* (2001) modeled the dynamic problem by using a displacement function that connects the new decision with the previous decision. Therefore, the decisions about landing time can take operational and environmental changes into account. In the related studies, the dynamic nature was normally addressed through the combination of two tech-

niques: scheduling window and freezing horizon, which are used for problems focusing on real time optimizations. The scheduling window refers to a fixed time interval. Aircraft in this interval can be scheduled to specific waypoints or runways. Freezing horizon is used for setting up a time threshold to exclude the aircraft that are no longer influenced by the dynamic issues. Beasley *et al.* (2004) defined a freezing horizon such that aircraft that are scheduled to land within this freezing horizon have their landing time frozen. Moreover, the scheduling window approach ensures continuous advance of the time horizon under consideration, in which new arrivals are included into the problem. Hu and Chen (2005) introduced the Receding Horizon Control (RHC) into Genetic Algorithm (GA) to solve the ASSP with a goal of minimizing the airborne delay and increasing runway throughput. RHC here is similar to the scheduling window approach. It is an optimization strategy that makes decisions by looking ahead for N steps in terms of a given cost/criterion. In this approach, only the decisions for the first step associated with this window are actually implemented. Further, based on the same problem, Hu and Di Paolo (2008) extended the previous work in the solution algorithm by designing an efficient GA based on a binary representation of arriving queues. Bencheikh *et al.* (2016) took advantage of the algorithm of Ant Colony Optimization (ACO) to tackle the ALP (see Bencheikh *et al.* (2011)) in a dynamic environment with two dynamic events: the appearance of new aircraft and the closure of a runway. In addition, Bennell *et al.* (2017) investigated both the static/off-line and the dynamic/on-line ALP. The dynamic problem is solved by periodically updating new arrival information based on the rolling horizon frame with an iterated descent algorithm or dynamic programming. A thorough computational evaluation is performed using data from Heathrow airport and randomly generated test data.

2.2.2 Aircraft arrival scheduling problem in the TMA

Most scheduling models in the literature consider the TMA as a single resource, typically the runway. However, many research works took the operation in the TMA into consideration. In this case, the sequencing and scheduling of aircraft are performed in a complex environment with controls such as holding, stacking, speed regulation, and route assignment. With a larger scope to consider, relevant resources of real route structure in the TMA for a specific airport can be studied and incorporated into the model. A series of studies based on the job-shop scheduling model were conducted by a research group. Bianco *et al.* (2006) has considered the airspace segments in the entire TMA and runways as machines and the aircraft with given earliest arrival times as jobs. The separation requirement is modeled as sequence dependent machine set-up times. In this study, CPS was also considered. The delay and the throughput were addressed in two separate models so as to verify the model performances. In the later work, additional real-world constraints such as holding time, time window constraint, resources capacity and blocking constraint at runways were taken into account. Branch and Bound (B&B) and a heuristic version of it were adapted for this problem (D'Ariano *et al.*, 2010). Further, D'Ariano *et al.* (2012) extended the previous work by adding a rerouting decision in the TMA. The algorithms of a truncated B&B and a Tabu Search (TS) combined with the neighborhood structures were applied and compared.

Apart from handling the problem by using the job-shop scheduling model, in recent

studies, model formulation based on the realistic route structures has also been applied. Eun *et al.* (2010) analyzed the real trajectory data to specify several waypoints from different merging routes as reference points in the TMA. Aircraft trajectories were adjusted by applying discrete delays at the reference points. Therefore, the safe spacing at the Final Approach Fix (FAF) can be guaranteed. The discrete delays were considered as decision variables and the objective of this work is to minimize the sum of delays. This problem is solved by using a B&B algorithm with LP and Lagrangian dual decomposition. Messaoud *et al.* (2018) established the model from a practical standpoint by considering decisions from air traffic controllers. This problem addressed the sequencing and scheduling of aircraft in the approach phase, taking into account the operations on the TMA resources (air segments) and on the runway. A detailed MILP formulation subject to the separation requirements and time window constraints was proposed, in which the objective is to minimize the maximum consecutive delay over all aircraft at the TMA entry points, at the FAF, as well as at the runway thresholds.

In addition, safety issues in the TMA can be emphasized by performing a conflict detection and resolution. Ma *et al.* (2019) has used network abstraction for the arrival routes. Longitudinal and horizontal separation losses are considered in a manner of flight by flight conflict quantification on the route resources. The objective is to minimize the weighted sum of the number of conflicts in the network and the delays. Nikoleris and Erzberger (2014) developed an autonomous system including trajectory prediction in the terminal area. Conflict detection and resolution were performed. The separation losses at the FAF were detected. Speed controls, horizontal maneuvers and altitude changes are used to resolve the potential conflicts.

The dynamic on-line problems in the TMA and at the runways have been studied as well. Murça and Müller (2015) used typical techniques of time based scheduling window and freeze horizon to solve the static sub-problems every fixed amount of time. In this work, the actual terminal arrival routes and related control actions were analyzed at a specific airport. A MILP model is formulated, in which standard and alternative predefined arrival routes and holding times are set as decisions. The objective function is established based on a penalty cost function, in which the delay caused by holding and route changes for individual aircraft are involved. In Samà *et al.* (2013), the rolling horizon approach is proposed. This approach enabled the dynamic management of aircraft ingoing and outgoing of the traffic control area. The target problem retained the framework of job-shop scheduling model which can be found from their previous work (D'Ariano *et al.*, 2010, 2012). A common feature of the dynamic optimization of the aircraft scheduling problem is that they enhance the static model by discretizing the time horizon so as to take into account the newly appeared aircraft and remove the aircraft outbound over time.

2.2.3 Arrival route structure management

Aircraft arrival management can benefit from not only flight schedule optimization but also arrival route structure design. The possible deployments of different solutions increase flexibility in the design of the TMA route layouts and landing procedures. One of the typical achievements is the development of Point Merge System (PMS) proposed by EUROCONTROL (EUROCONTROL, 2019). PMS replaces the traditional standard

arrival route structure with a more flexible one that consists of a point (the merge point) and predefined legs which are equidistant from the merge point. Aircraft on the legs are directed to the merge point at appropriate times, from which controllers can have a clear view of arrival sequence. It is worth mentioning that the technique of Continuous Descent Operation (CDO) has been proven to be very promising in the aircraft arrival segment with one limitation, which is the application of this approach in a busy time period may cause significant delay and trajectory deviation (Clarke *et al.*, 2004; John and Kamgarpour, 2010). However, PMS enhances aircraft operation performance by allowing the application of CDO for flights in a high-density traffic environment at a lower altitude. The benefits of this combination are verified in terms of fuel saving, emission reduction and noise control. Most of the literature related to the aircraft arrival sequencing and scheduling problem with PMS are focused on performance validation. Boursier *et al.* (2007) compared the performances of the PMS with the current arrival control methods. Meric and Turan (2016) emphasized the advantages of PMS by describing the differences between the CDO and the traditional vectoring. Usanmaz *et al.* (2019) conducted an assessment based on the real-time implementation of the PMS at Ankara Esenboga International Airport in terms of the number of instructions and the flight occupation times required from the controller. Optimizations for the actual flight operation in the PMS system have been conducted. Liang *et al.* (2018) addressed the ASSP by establishing an autonomous PMS framework based on Beijing Capital International airport.

Even though PMS is one of the aviation block upgrades and provides very promising outcomes, its application is not yet compatible with all airports. Therefore, studies that make progress based on the current arrival procedures still hold great value in the arrival sequencing and scheduling problem. Zuniga *et al.* (2011) proposed an approach to address conflict detection and resolution problems when aircraft density increases by improving the airspace utilization in the TMA. Aircraft speed control and route path changes are performed, where the alternative paths form a trapezoid with the original route by sharing the beginning and endpoint of the specific route segment. In the follow-up study, Zuniga *et al.* (2013) further elaborated the model by incorporating the path shortening and stretching technique. Fig 2.1 illustrates the simple example of the PMS and the route stretching strategy proposed in Zuniga *et al.* (2011).

2.3 Air traffic optimization under uncertainty

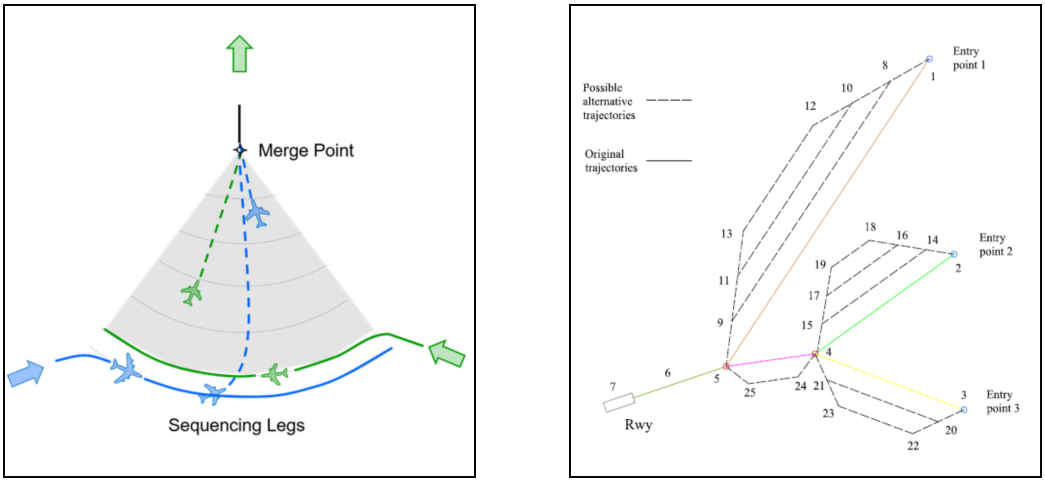
The above-mentioned studies are optimized or simulated based on exact information. Therefore the sequencing and scheduling are carried out in an ideal situation where aircraft follow all the instructions precisely. However, in real operation, uncertainties such as aircraft maneuvering, weather effects, information delay etc. may lead to the fact that the predicted flight information and operational conditions in the airspace or at the airport vary with time.

Many works were dedicated to better understanding the pattern of prediction errors during the flight operation (Tielrooij *et al.*, 2015). While the fact is that as a general term for all kinds of errors, the statistical attributes of the uncertainties being analyzed are highly dependent on the available information in multiple aspects such as airspace configuration, weather condition, traffic density or human factors. Tielrooij *et al.* (2015)

used the Flight Update Message (FUM) that indicates the expected arrival time of flights at waypoints to estimate the prediction errors. In the work of Wanke *et al.* (2004), prediction error of sector capacity evaluation was estimated. The predicted demand estimated in each sector was provided by the Enhanced Traffic Management System (ETMS). Then, the actual history of flight information was used as the reference for deriving the prediction error. Mercer *et al.* (2013) investigated the wind forecast error and aircraft performance error based on the information provided by the decision support tool. All these works demonstrated that uncertainty is a critical issue inherent to air traffic management that needs to be dealt with.

Typical approaches used for uncertainty optimization include stochastic programming, robust optimization, chance constrained optimization and distributionally robust optimization (Li and Ierapetritou, 2008; Sahinidis, 2004).

- **Stochastic Programming:** The most common applications of stochastic programming are the two-stage or multi-stage stochastic problems (Bosson *et al.*, 2014; Khassiba *et al.*, 2019), where one assumes that the uncertain parameters can be represented by the known probability distributions based on a complete knowledge of the uncertainty. The selection of the tackled problems for the stages are usually based on the prioritization of the particular information required. That is to say, for a two-stage problem, variables that provide prior information before the realization of uncertainty parameters have to be decided in the first stage. Further, corrective measures or recourse are activated at the second stage to mitigate effects from uncertainty and avoid infeasibilities. The general process is to take decisions for the first stage such that the total cost of the first stage and the expected cost of the second stage are minimized. Due to uncertainty, the cost of the second stage can be considered as a random variable, and the expected function is introduced for the second stage. One broadly used method that approximates the expectation function by averaging the values obtained from a finite number of scenarios is the



(a) PMS (EUROCONTROL, 2019). (b) Route stretching (Zuniga *et al.*, 2011).

Figure 2.1: Simple Example of PMS (EUROCONTROL, 2019) and the route path stretching technique used in Zuniga *et al.* (2011).

Sample Average Approach (SAA) method. In Kleywegt *et al.* (2002), the implementation and performance of SAA method based on the Monte-Carlo approach has been comprehensively studied from both theoretical and experimental aspects for a stochastic optimization problem. The accuracy of stochastic optimization is based on the expansion of scenario exploration. While, as the problem size gets larger, the number of random variables increases, which drastically yields increased scenarios. Consequently, it is computationally intensive for a high dimensional problem.

- **Robust optimization:** In robust optimization, the probability distribution of the parameter representing the underlying uncertainty is normally unknown. The concept of robust optimization is to define the reasonable uncertainty set, within which the optimized solution is computed (Ben-Tal and Nemirovski, 2002). Robust optimization is a conservative method that focuses on immunizing against the worst case or scenario which are derived from all the possible input data (Ben-Tal and Nemirovski, 2002; Bertsimas and Sim, 2003). One of the common uses of robust optimization in dealing with the arrival management related problem is the robust min-max regret method. This method tries to minimize the maximum regret value, namely, the difference of objective values between a made decision and the optimal decision obtained from the worst scenario (Ng *et al.*, 2017). Kapolke *et al.* (2016) implemented both the robust optimization model and the stochastic optimization model for a runway scheduling problem. An explicit comparison between these two models in terms of model formulation and performance was provided.

- **Chance constrained optimization:** According to different problems and designs, chance constrained optimization can sometimes be classified into the field of stochastic programming (Marla *et al.*, 2020) or robust optimization (Esche *et al.*, 2019). As the name indicates, chance constrained optimization allows for the violation of some inequalities by introducing a desired probability level (Charnes and Cooper, 1959). In Chen *et al.* (2017), a chance constrained optimization approach is established by including probabilistic sector capacity constraints for the large scale air traffic flow management problem. With this method, a less conservative solution can be obtained compared to the robust optimization approach while maintaining a certain level of solution quality.

- **Distributionally robust optimization:** The distributionally robust optimization emphasizes the ambiguity of the probability distributions of the uncertain parameters. Analogous to robust optimization that considers the worst-case parameter realization in the uncertainty set, distributionally robust optimization seeks to protect against the worst-case with ambiguous probability distribution. Under this case, the probability distribution for the uncertain parameter is unknown but belongs to an infinite set of predefined probability distributions (Rahimian and Mehrotra, 2019).

Among multiple forms of robust optimization and stochastic programming, interpreting the uncertainty for probability related parameters sometimes is necessary, which indicates the wide implementation of probabilistic modeling. An abundance of studies focus on the air traffic flow management problem in the enroute segment by applying probabilistic interpretation for the time information and then the expected number of flights is estimated

for specific sectors (Gilbo *et al.*, 2011; Caron *et al.*, 2013; Gonze *et al.*, 2018). Uncertainty can also be managed by imposing separation buffers between the consecutively operated aircraft (Meyn and Erzberger, 2005; Heidt *et al.*, 2016). Scala *et al.* (2019, 2021) aimed at reducing the impact of uncertainty by enlarging the separation minima under a simulation optimization framework by performing the conflict detection and resolution. Intuitively speaking, inserting separation buffers with respect to the nominal separation minima is a very effective way to generate a robust solution. However, regarding the growth of traffic demand, fixed buffering limits airspace capacity, reduces the flexibility of aircraft scheduling, and therefore may introduce a more complex problem.

2.4 Extended AMAN

The conception of E-AMAN is one of the SESAR solutions. The main goal of E-AMAN is to enhance the cooperation of the traffic in the TMA to the further upstream management. By implementing an earlier enroute speed control while considering the operational constraints in near airport airspace, a smooth and efficient arrival process can be achieved. Fig 2.2 shows the geographical extension of a live trail carried out for the arrival traffic flow of south-east bound at the Paris Charles de Gaulle (CDG) airport. In this case, the management horizon has been extended up to 350NM. The decision of the horizon extension depends on the context in which E-AMAN is applied. The possible control range can be extended up to 500NM with an associate time horizon up to 2 hours.

Since E-AMAN is a new strategical concept, limited studies can be found and most of them are based on the controllers' decision control rules. Jones *et al.* (2013) presented a dynamic multi-objective integer programming model incorporating the main objectives of fuel saving and delay minimizing at the airport. In this paper, practical control manners

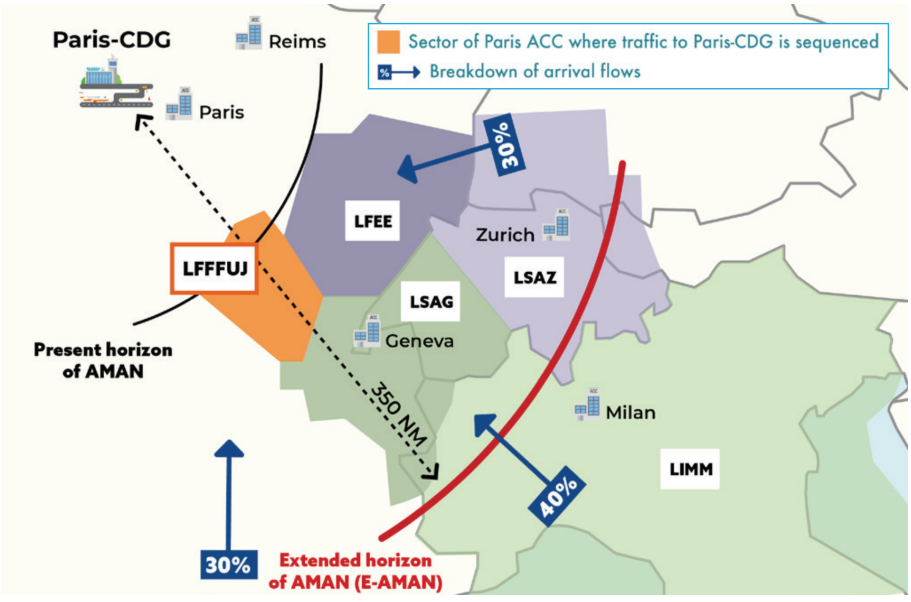


Figure 2.2: Live trials in 2018 on cross-border arrival management for South-East Paris CDG inbound flows. AMAN horizon is extended up to 350 NM involving a collaborative process between Paris, Reims, Geneva, Zurich and Milan ACC SESAR (2019*a*).

are considered for a horizon of 500NM from the airport. Aircraft are controlled by the initial Estimated Time of Arrival (ETA) and the updated Controlled Time of Arrival (CTA) which is issued depending on their time distances from the concerned horizon. For determining the priority of specific controls movements, two successive rolling horizon periods (scheduling window) were introduced. The objective in the most recent window is assigned with higher weights than the other window. Through this way, aircraft near the airport were urged to be compliance with the CTA while the later aircraft were preferred to be delay in the enroute segment. Then, a series of trail studies were conducted, which proved the effectiveness of this model. Peters (2017) introduced a practical control tool based on the E-AMAN concept and the trajectory information provided by Air Navigation Service Provider (ANSP). Enroute speed adjustment and alternative route assignment were applied to meet the Required Time of Arrival (RTA). Solution has been validated through live trials, and benefits in terms of delay reduction and emissions can be achieved.

Simulation based model is widely used for the validation of the E-AMAN conception. Swenson *et al.* (2011) proposed an advanced air traffic control decision support tool that integrates precise time and trajectory prediction. The vectoring in the meter fix, merge points and runways is conducted based on the sequence determined by an extended further upstream management. Delay are pre-considered as it can be absorbed in the cruising phase. This approach is implemented in the real operational environment and compared with current Air Traffic Control (ATC) operations under several levels of demands. Nikoleris *et al.* (2012) investigated the distance required for flights to absorb a certain amount of delay by clustering historical flight data that has been implemented with enroute speed control. By comparing with a simulation based delay-free schedule, the speed control strategy can absorb a considerable amount of delay with a margin of 10% from the nominal speeds. The delay transferring or sharing strategy embedded in the E-AMAN concept relies on speed control for an extended distance. As long as the relevant controls can be effectively delivered to the pilots, it is quite flexible in its application scope. Delgado has conducted a series of researches aiming at discovering the benefits of speed control in the cruising segment for fuel saving (Delgado and Prats, 2009) and delay absorption (Delgado and Prats, 2012). An extended study combining the Ground Delay Program (GDP) with the cruising speed regulation is proposed to recover the delay that is supposed to be imposed to GDP. This strategy provides alternative or back up options to balance the capacity in the airport and in the air (Delgado *et al.*, 2013).

2.5 Solution algorithms

Most of the problems mentioned above are solved through mathematical optimization. The algorithms implemented for the ASSP can be categorized into exact approach and heuristic approach. The choice of the solution algorithm is dependent on the characteristics of the addressed problem. The ultimate goal is to find an optimal solution or near optimal solution within an acceptable time. Here, the two types of algorithms are discussed.

2.5.1 Exact approaches

The exact approaches always solve an optimization problem to optimum. Therefore, this kind of approach is broadly used in the literature. In this part, we present the algorithms that are most commonly applied:

- **Mix Integer Linear Programming (MILP)**
  Strictly speaking, MILP is a formulation method. It can be solved using solvers such as CPLEX, Gurobi, etc. in an exact manner. Many above-mentioned works formulated the optimization models based on the MILP (Beasley *et al.*, 2000; Heidt *et al.*, 2014; Samà, D'Ariano, D'Ariano and Pacciarelli, 2017), which presents its advantage of great flexibility in modeling and allows to easily introduce the desired extensions (such as the permissible landing time window and the minimum separation time). Many works have used the commercial MILP solver (typically IBM-CPLEX) to formulate and solve the aircraft scheduling problem (see. e.g. Beasley *et al.* (2004); Messaoud *et al.* (2018); Diallo *et al.* (2012)). However, one limitation is that only small instances can be addressed. The performance of this approach deteriorates for large instances.

- **Dynamic Programming (DP)**
  This method defines the relationships between problem states by breaking up the problem into sub-problems (sub-stages). Thus, the problem can be solved by recursion or in a one by one divided manner. Using DP, the scheduling and re-scheduling can be easily taken into consideration when new events or updates of data are obtained.

  Taking advantage of the features of this algorithm, Balakrishnan and Chandran (2006) implemented dynamic programming to solve the aircraft scheduling problem at the runway. This problem is interpreted as a modified shortest path problem on a growing network that consists of stages indicating the current possible landing sequence. In this way, the shortest path in terms of operation time can be generated under multiple constraints such as landing time windows and separation requirements. The same solution algorithm has been employed in continued studies that incorporate additional constraints or modified objectives (Chandran and Balakrishnan, 2007; Lee and Balakrishnan, 2008; Balakrishnan and Chandran, 2010). Furini *et al.* (2014) proposed a fast DP algorithm based on the algorithm proposed in Balakrishnan and Chandran (2010) to minimize the sum of weighted deviations under the CPS constraint. An additional state space reduction technique involving a heuristic searching for an upper bound and a lower bound with respect to the completion time is embedded into the classical DP algorithm. Lieder *et al.* (2015) resolved the ALP on multiple independent runways based on general assumptions (multiple runways, positive target times, and limited time windows). This work focused on computational complexity reduction and developed an efficient DP based on the framework proposed in Briskorn and Stolletz (2014). In this study, the DP algorithm was applied with a dominance criterion which is used for state space reduction while searching for the optimal solution. The dominance relation is established between two states. First, at least the same number of aircraft with a

specific class has been scheduled. Second, the landing time deviation cost of the dominated state does not exceed the other. Third, the landing time of each aircraft should be earlier in the dominate state than the dominated states. By removing the dominated states at each step, the state space search is more efficient, which significantly reduces the computational burden when solving the ALP with multiple runways. Numerical comparison shows that the DP approach yields optimum schedules significantly faster than a standard MIP solver (CPLEX). Further, Lieder and Stolletz (2016) implemented the same DP for realistic runway scheduling problems at London Heathrow airport and Frankfurt airport.

- **Branch and Bound (B&B)**
  This is a classical algorithm that consists of step-wise enumeration of possible solutions. The exploration of the entire state space follows a tree structure with nodes corresponding to the sub-problems, and the leaves correspond to the solutions. Depending on the bounds associated with the nodes, the sub-problems may be refined into smaller sub-problems or aborted.

  Bianco *et al.* (1978) uses B&B algorithm to solve aircraft sequencing problem inside the TMA and at one runway. Moreover, Ernst *et al.* (1999) also applied the B&B algorithm and a space search heuristic to solve ALP on a single runway and on multiple runways. A specialized simplex algorithm was used to rapidly evaluate the landing time. By combining the two approaches, the computational efficiency has been increased as well. Starting from D'Ariano *et al.* (2010), a series of works carried out by this research group have applied B&B algorithm for the problems of scheduling and routing, landings and take-offs, taking into consideration the interdependent runways and air segments in the TMA. The models are well developed by incorporating more practical control constraints and possibilities. Further, some works carried out the comparison of the B&B algorithm and other heuristic approaches (D'Ariano *et al.*, 2012, 2015; Samà *et al.*, 2014; Samà, D'Ariano, Corman and Pacciarelli, 2017).

The exact solution approaches require a relatively long time to solve an optimization problem, therefore, it is not suitable to be implemented in reality considering that the air traffic is operated in a highly varied environment and frequent computations are required.

2.5.2 Heuristic approaches

Compared to the exact algorithms, though the heuristic algorithms do not ensure a globally optimal solution, they can improve the computational efficiency to a great extend. Considering that all kinds of uncertainties make the optimal solution vulnerable, algorithms that provide fast and near-optimal solutions gain much more interest. From the literature, there are several meta-heuristic algorithms that are commonly used for the Aircraft Scheduling Problem (ASP)and ASSP.

- **Genetic Algorithm (GA)**
  This algorithm aims to evolve a population of solutions through several steps, including selection, crossover, and mutation during a number of iterations.

In the early study, Ciesielski and Scerri (1998) provided two genetic algorithms: the standard binary genetic algorithm and a "seeding" modification algorithm that iteratively updates the initial population for the dynamic case. In Hu and Chen (2005), a typical GA with only mutation operator was employed for the ASSP at the runway. Then, in Hu and Di Paolo (2009), the potential inefficiency of the existing GA were analyzed. By employing the uniform crossover operator to two kinds of mutation mechanisms, a high efficient GA was proposed. In both works, the authors combined the proposed GA with the RHC concept, which further facilitated the implementation of the algorithm for the ASSP on multiple runways in real life situations. In Kupfer (2009), GA has been embedded in a greedy algorithm to provide better performance in terms of computational efficiency while maintaining the solution quality. It is worth mentioning that Non-Dominated Sorting Genetic Algorithm (NSGA) has been broadly used in the multi-objective optimization problem for generating a Pareto Frontier (PF). The PF shows the controversial objectives under equality constraints at different levels (Tang *et al.*, 2008; Caron *et al.*, 2013), which enables the stakeholders to make trade-offs in the solution set indicated in the PF. Actually, GA is a population-based algorithm. When solving a problem that involves a large-dimensional state space, a huge memory is required. Therefore the population based algorithms have their limitations.

- **Ant Colony Optimization (ACO)**
  This is a class of optimization algorithm modeled on the actions of an ant colony, which is initially used for solving the Traveling Salesman Problem (TSP). Therefore, the problems that need to be solved are normally converted into the shortest path finding problem on a weighted path in a graph. With this feature, ACO is also very promising in dealing with the problem with dynamic nature.

  Works that apply ACO for the ALP can rely on different problem interpretations. First, a graphical representation of the problem can be established. Bencheikh *et al.* (2011) defined a two-level graph with the available runways in the first level and aircraft in the second level. The ants first selected the runway based on predefined probabilistic rules taking into account the number of aircraft associated with this runway and the aircraft separation requirements. Then, the ant continued to select the aircraft in accordance with the priority of aircraft and the penalty cost of each aircraft. Further, Bencheikh *et al.* (2016) addressed a dynamic ALP and took advantage of the constructive feature of ACO to solve the problem. Another straightforward interpretation suggested that the ALP can also be transferred into a TSP, in which the runways were regarded as travelers and the aircraft are considered as nodes (Xu, 2017). In order to increase computational efficiency, techniques can be adapted in the searching process of the ACO. Zhan *et al.* (2010) has combined the RHC with the ACO to solve the ASSP. Then, the ACO is further enhanced by applying a heuristic local search, in which the new solution is obtained by exchanging the position of two aircraft. Final results suggested that even without the rolling horizon technique, the ACO that accommodated the local search process outperforms both the GA and the basic ACO.

- **Tabu search (TS):**

This is a meta-heuristic local search algorithm that performs the procedure of prohibiting the already visited solutions through user provided rules by putting the recently visited solutions in the tabu list. D'Ariano *et al.* (2012) addressed an aircraft conflict detection and resolution problem and included a new re-routing strategy in the decision. The re-routing problem was solved by TS, considering the initial scheduling problem without re-routing decision. The advantage of TS in reducing delays within a short computation time was demonstrated.

- **Simulated annealing (SA)**
  This algorithm relies on a stochastic global search, in which a worse solution can be accepted with a probability. This probability is computed based on a function involving temperature and the quality difference between the candidate solutions. The randomness makes SA appropriate for nonlinear objective functions where other local search algorithms do not operate well.

  Câmara *et al.* (2016) has compared two meta-heuristic algorithms: SA and the TS for ALP. Experiments show that SA outperforms TS by up to 55% and FCFS by up to 19%. Liang *et al.* (2018) addressed the aircraft sequencing and scheduling problem in the terminal area based on the PMS. The complexity of the problem has been solved by using a simulation and optimization framework. The study has applied SA and hill climbing algorithm to solve the problem. Results showed that the hill climbing is faster than SA but the quality of the solution is less satisfied than SA. Ma *et al.* (2019) addressed the aircraft scheduling problem for both arrival and departure in the TMA and at the runways. They incorporated a roulette wheel search strategy in the SA to achieve a high efficient optimization.

In addition to the mentioned algorithms, there are other algorithms have been applied to solve the aircraft scheduling problem, such as Greedy algorithm (Amrahov and Alsalihe, 2011), Scatter search (Pinol and Beasley, 2006), etc. Some studies focused on comparing several methods on the same problem to identify the most effective one. Bennell *et al.* (2017) applied the typical approaches including DP, iterated descent, SA and the FCFS to tackle the aircraft landing problem in both static and dynamic environments. A multi-objective function is established, which consists of average landing time, total penalty for violating time windows and fuel consumption. Further, the dynamic problem was tackled. Results suggested the advantage of the heuristic approach in handling the large-size problem. Samà, D'Ariano, Corman and Pacciarelli (2017) dug deeper and broader in the direction of algorithmic improvements for both exact and meta-heuristic algorithms based on the MILP formulation proposed in Samà *et al.* (2014). Câmara *et al.* (2016) has compared the exact algorithm: FCFS, and the meta-heuristic algorithms: SA and the TS for ALP. Experimental results are numerically shown to present the performance of each algorithm.

Other studies hybridize several methods in order to enhance the performance. Ernst *et al.* (1999) tackled the classical ALP. For solving this problem, a simplex algorithm is firstly used. This algorithm is later used to specify the lower bounds of the B&B algorithm. Faye (2018) proposed a SA approach combined with the exact LP method to solve the classic ALP on a single runway case. The objective is to minimize the total delay cost taking into account constraints such as time windows, separation constraints

and predefined sequence options. The author chose two exact algorithms: the DP and the LP. Both of them are thoroughly explained and then embedded in an iterative algorithm based on a SA framework. Final results suggested that the proposed DP model is at least 7.8 times faster than the LP method, which also helps in reducing the number of iterations for finding high quality solutions. Bencheikh *et al.* (2013) described four hybrid algorithms to solve the ALP proposed by Beasley *et al.* (2000) in a multiple runway case. Hybrid algorithms were proposed by incorporating TS and GA. Through manipulating the application stages of the two heuristic methods and the algorithms themselves, four hybrid algorithms were proposed and the results indicated that each of the algorithms has its own advantages in different aspects.

In our work, we have developed several implementations of the SA and the next section summarizes the main features of this meta-heuristic approach.

2.6 Simulated annealing algorithm

The main advantage of SA is its simplicity of application and adaptation. In this section, the details of this algorithm are described.

Basic of simulated annealing

In the early 1980s, Kirkpatrick *et al.* (1983) introduced SA in the field of solving the combinatorial optimization problem. The SA algorithm is essentially inspired by the physical annealing of materials. This process involves bringing a solid to a low energy state after raising its temperature. At high temperature, the particles are in a random state. A slow cooling process enables the solid reaches a well ordered state of minimum energy. If a sudden cooling occurs, the material is found with metastable structures. Those two types of cooling are displayed in Fig. 2.3.

The algorithm includes the heating and the controlled cooling process. An important control parameter in this optimization process is the temperature. The heating process explores the solution space to search for a temperature that can ensure a sufficient and deep exploration of state space for the cooling process. The cooling process refers to the decreasing of the temperature which can be interpreted as a slow reduction in the probability of accepting worse solutions while exploring the solution space. In order to achieve either the heating or cooling process, several important factors that define the exact evolution of the states are demonstrated as follows:

- **Temperature:** The temperature in the SA can be interpreted as an indicator to show optimization status. It can be given as $T_k = f(T_{k-1})$ for consecutive iterations, where k denotes the finished number of iterations and $f(T)$ is a cooling function set according to the targeted problem. By cooling the temperature slowly the approximate global optimum will be found.

- **Neighborhood solution:** During the optimization process, we have the current state i corresponding to an objective value E_i. The next new state j should be chosen from a design regarding the given circumstance. It can be chosen from the state space either by randomly changing a decision or by using some auxiliary

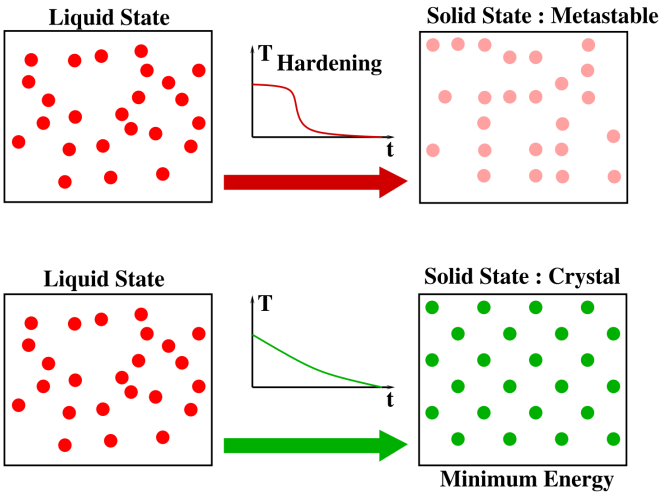


Figure 2.3: The process of the physical SA of material in terms of energy changing. When the temperature is high, the material is in a liquid state (left). For a hardening process, the material reaches a solid state with non-minimal energy (metastable state; top right). In this case, the structure of the atoms has no symmetry. During a slow annealing process, the material also reaches a solid state, for which atoms are organized with symmetry (crystal; bottom right) Delahaye *et al.* (2019).

techniques. Then, the objective value E_j corresponding to j can be evaluated to decide whether this neighborhood solution will be accepted or not.

- **Acceptance probability:** The acceptance of the neighborhood solution is based on the Metropolis algorithm (Metropolis *et al.*, 1953). If the difference $E_i - E_j$ is positive, the state j becomes the new current state. Otherwise, if the difference is less than or equal to 0, then the transition from state i to state j is based on a random probability Pr which is expressed as:

$$Pr = \begin{cases} 1, & \text{if } E_j < E_i \\ e^{\frac{E_i - E_j}{K_b \cdot T}}, & \text{otherwise.} \end{cases} \tag{2.1}$$

- **Transition:** A transition is defined as the replacement of the current solution by a neighborhood solution. It consists of two main steps: the neighborhood generation and acceptance or rejection of the neighborhood solution. For each temperature, a certain number of transitions is specified to let the algorithm explore the state space.

- **Iteration:** Each iteration corresponds to a specific temperature, for which an amount of transitions are carried out.

The functional paradigm for the SA is summarized in Algorithm 1, where N is the number of transitions for each temperature.

During the optimization process (cooling process), the number of transitions at each temperature and the initial temperature should be initialized. The initial temperature can be determined by the heating process which is similar to the cooling process except that the temperature is continuously increased. In the heating process, the initial temperature

Algorithm 1 Simulated annealing algorithm.

Require: Initialization: number of transitions for each iteration N, initial temperature T_0, random number $\epsilon \in [0, 1]$;

E_0=ObjectiveFunction(x_0);
$x_i \leftarrow x_0$;
$E_i \leftarrow E_0$;
while $T > 0$ or x_i is not optimal solution **do**
    for $k = 1$ to N **do**
        x_j =NeighborhoodGeneration(x_i);
        Calculate the objective of x_j, E_j=ObjectiveFunction(x_j);
        if $E_i > E_j$ **then**
            $x_i \leftarrow x_j$;
        else if $\epsilon < \exp(\frac{E_i - E_j}{T})$ **then**
            $x_i \leftarrow x_j$;
        end if
    end for
    T =TempratureDecrease(T);
end while
return x_i

is fixed when the acceptance rate of about 80% is reached (this means 80% of the proposed neighborhood solutions are accepted). The initial state x_0 is usually randomly selected. A high number of transitions ensures that the state space can be explored sufficiently at one temperature. In the acceptance procedure, although the better solutions are preferred, we also accept worse solutions to avoid getting stuck in a local optimum. The cooling process can be intuitively seen in Fig. 2.4. Note that objective values are decreasing in a general trend, while some hill-climbing moves representing the acceptance of the worse solution appear. Moreover, the decreasing of the temperature makes the probability of accepting worse neighborhood gets lower and the global approximate optimal solution will be finally found.

Practical issues of SA

When one wishes to implement the SA algorithm for its particular application, the detailed definition for each step in the algorithm needs to be specified to ensure the adaption of the algorithm. The common designs corresponding to the main steps are introduced.

- **Temperature evolving pattern:**
  The cooling schedule parameters including start temperature, cooling function, termination conditions are closely related to the rate at which SA can obtain an optimal solution. The cooling function that describes how SA reduces the temperature to the next one is especially important. Four commonly used cooling patterns are introduced in Peprah *et al.* (2017).

  – Geometric law:

$$T_{i+1} = \beta_g T_i \tag{2.2}$$

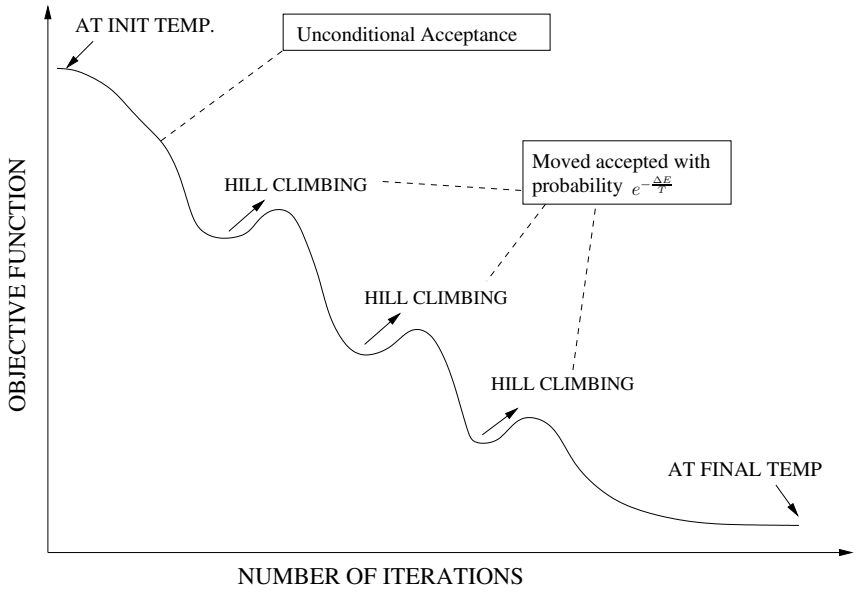


Figure 2.4: The cooling process of SA with respect to the objective value variation and the temperature changing.

where β_g is a cooling factor which is supposed to be a fixed value that can achieve a small decreasing rate of the temperature. Normally one can set the range of $\beta_g \in [0.8, 0.99]$, while no absolute rule is imposed for the value of β_g.

- Linear law:

$$T_{i+1} = T_i - \beta_l, \quad 0 < \beta_l \tag{2.3}$$

where β_l is a temperature reduction parameter, with a recommended range of $[0.1, 0.2]$.

- Logarithmic law:

$$T_{i+1} = T_0/log(i) \tag{2.4}$$

where T_0 is the initial temperature.

- Algorithmic-Geometric:

$$T_{i+1} = T_i\beta_{ag} + \eta_{ag} \tag{2.5}$$

This is a non-monotonic cooling schedule that combines the geometric and the linear law, β_{ag} and η_{ag} represent the cooling parameters for the geometric part and the linear part, respectively.

There are several ways to determine the stopping criteria.

- Upper limitation of the number of iterations (number of temperature changes).
- Upper limitation of the execution time of the algorithm.
- When a solution achieves the desired objective.
- When there is no improvement on the objective value for a predefined number of iterations or predefined execution time.

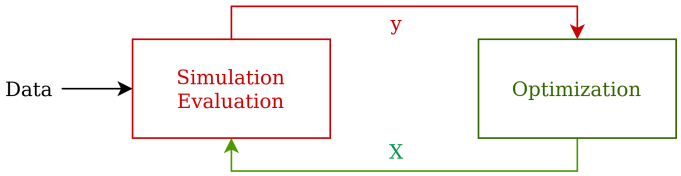


Figure 2.5: Objective function evaluation based on a simulation process (Delahaye *et al.*, 2019), where simulation is applied for obtaining the objective value y of the associated solution.

It is possible to have more than one kind of stopping criteria in the optimization procedure.

- **Neighborhood selection strategy:**
  In SA, the neighborhood selection strategy should be defined based on two factors: the *effectiveness* and the *efficiency*. The *effectiveness* refers to the capability of accessing all solutions in the state space and the *efficiency* refers to the fastness of locating a high quality neighborhood. A neighborhood selection strategy can be evaluated from the following elements:
  - **Neighborhood covering rate:** The number of neighborhoods that can be accessed from the current solution.
  - **Neighborhood generating time:** The time used for generating a neighborhood.
  - **Information volume:** The information used for performing guidance to high quality solutions.
- **Simulation based evaluation:**
  In many optimization applications, the objective function is evaluated by a computer simulation process. In such a case, the optimization algorithm controls the vector of decision variables X to produce the associated performance y through the simulation process, as shown in Fig. 2.5.

  Population-based algorithms such as GA is hard to be adapted to address such problems. Since for generating a new neighborhood, the simulation environment requires a huge amount of memory space, therefore this kind of solution algorithms have a high computational burden to solve the real-life problems. In contrast, as a single-solution based algorithm, SA gains more interest for real-life problems of high dimension.

  When we use a single solution based algorithm, the simulation environment is initiated for solution evaluation. The decisions corresponding to the current solution are put in the simulation environment to generate performance of this solution. This performance (i.e. objective value) is then transferred to the optimization algorithm. Afterwards, the second solution is chosen by the predefined neighborhood selection strategy and evaluated by the simulation process. It is worth mentioning that different solutions sharing the same simulation environment may lead to confusion if the solution information is not completely replaced in the simulation environment.

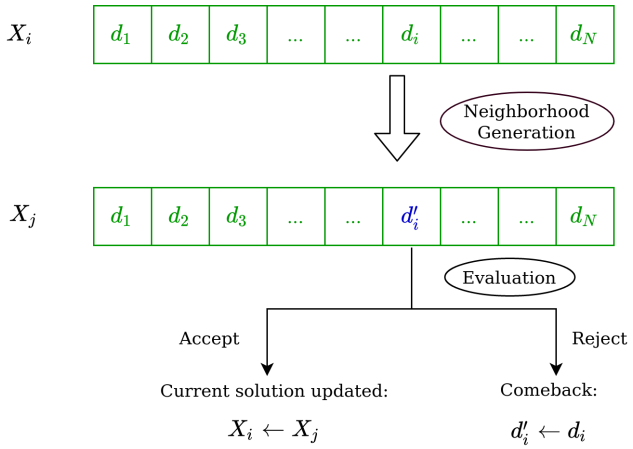


Figure 2.6: Neighborhood selection, evaluation process and the usage of the comeback operator for one transition in the SA algorithm. The comeback procedure is applied when the neighborhood is not accepted, and minimum memory is required by using this technique.

Therefore, the simulation environment must be initialized before evaluating the second solution. In this case, the memory space is no longer an issue, but the evaluation time may be excessive due to the reset of the simulation environment at each evaluation.

In the standard SA algorithm, a copy of the state space points (i.e. solutions in the state space) are requested for each transition. In fact, a state X_j is generated from the current state X_i through a copy in the memory of the computer. When we have a large state space, implementing such a copy at each transition may be inefficient and may drastically reduce the performance of SA. Therefore, we consider a come back operator, which moves back the solution to the state before the application of the neighborhood operator. In the simulation environment, only the relative changes on the neighborhood operator are updated, but not the whole situation. Let G be the neighborhood generation operator which transforms a point from X_i toX_j. The comeback operator is the inverse of the generation operator. The comeback operator is used under the condition that the neighborhood solution is rejected. Otherwise, we apply the aforementioned procedures of clearing the information for the current solution in the simulation environment, evaluating the neighborhood solution and updating the current solution successively. In Fig. 2.6, the exact process is illustrated.

Considering the advantages of SA and the problem we are going to solve, SA has been chosen as the main solution algorithm to be applied in a simulation based optimization framework, different techniques are combined with the generic SA to further enhance its performance.

2.7 Conclusion

In this chapter, we reviewed several kinds of optimization problems related to arrival air traffic in the terminal area and at the runways. First, the ASP for landing runways are reviewed. This problem is referred to as the ALP, which is one of the most commonly studied problems for the management of arrival air traffic. The ALP aims to increase the efficiency and capacity of the runway system while accommodating various operational constraints such as wake turbulence separations, flight time window, precedence constraints etc. The objectives of the problems that represent the interests of different stakeholders include throughput, delay and environmental effects etc. Previous research works are separately discussed in terms of the number of objectives that are considered in the problems.

Afterwards, the studies related to the airport surrounding area are focused. Compared to the ALP, the ASP in the TMA requires detailed information for model formulation. Specifically, information such as the air segments or arrival routes have to be considered. In addition, the topology of arrival route is presented as well. Advanced route structure design such as PMS is introduced.

In the literature related to optimization of air traffic management, some researchers focused on operational uncertainty. Different kinds of uncertainty optimization methods are applied in accordance with their own problem contexts. Typical approaches used for uncertainty optimization are introduced.

Recently, an advanced concept of E-AMAN that aims at enhancing the cooperation between the control in the TMA and the upstream control center has been introduced. The key benefits of this concept include improving control efficiency, reducing holding, fuel costs and lowering emissions by conducting earlier sequencing and speed regulation in the enroute phase. Studies related to this concept or applying the main strategy of this concept are presented.

Finally, the solution algorithms commonly employed in the literature for arrival aircraft sequencing and scheduling optimization problems are presented. Some works compared the exact algorithms and heuristic approaches, which proved that with complex problems, heuristic algorithms outperform the exact algorithm in multiple aspects. Among all of them, we choose SA as the solution algorithm for our studies considering its efficiency and high adaptability.

Chapter 3

Arrival air traffic optimization in the terminal area under uncertainty

Arrival MANagement (AMAN) involves proper controls of arrival traffic with the help of the decision support tool. In this process, all factors that have an effect on aircraft operations should be considered, among them, uncertainty is a critical element that needs to be properly dealt with. In this chapter, we address an arrival aircraft scheduling optimization problem under uncertainty. Section 3.1 introduces the problem and the mathematical models including a deterministic model, a model incorporating separation buffer and a single stage stochastic model. Section 3.2 describes the solution approach that we use to tackle our problem. Then, in Section 3.3, a simulation framework is applied to investigate the performances of the presented models under disturbances. In Section 3.4, the computational results are illustrated, and the simulation results obtained based on the optimized solutions from the three models are compared in terms of conflict absorption ability when uncertainty arises. Finally, Section 3.5 concludes this chapter.

3.1 Problem description and mathematical modeling

In this section, we describe three air traffic optimization models considering the air traffic operation in the Terminal Maneuvering Area (TMA). Flight input data are given and network abstraction is described. We first introduce a deterministic model by specifying the decision variables, constraints and objective function. Based on the model framework of the deterministic model, uncertainty is incorporated into the proposed model. In addition, a benchmark model that incorporates separation buffers is introduced. In accordance with the characteristics of each model, different objective functions are given respectively.

3.1.1 Problem overview

Flights that are intended to land at an airport are required to arrive at the designated entry points of the surrounding airspace of this airport at a specific time. Flight flows from multiple directions merge at certain points to be sequenced and then proceed to

the designated runways. During this process, the schedule should be carefully determined according to separation requirements, the runway acceptance rate, and the unexpected uncertainty etc. Focusing on the above issues, we address the arrival scheduling problem in the TMA and runways under uncertainty so as to provide a robust solution that can avoid potential risks.

3.1.2 Network Structure

Similar to the work of Ma *et al.* (2019), we model the TMA arrival network as a graph $\mathcal{G}(\mathcal{N}, \mathcal{L})$. The node set $\mathcal{N}$ is the set of waypoints and the link set $\mathcal{L}$ is the set of links interconnecting these waypoints with straight-line segments. We assume that $\mathcal{N} = \mathcal{N}_e \cup \mathcal{N}_w \cup \mathcal{N}_r$ where $\mathcal{N}_e$ contains the entry points of the TMA, $\mathcal{N}_r$ represents the runway thresholds, and $\mathcal{N}_w$ is the set of waypoints used to connect the entry points to the runway thresholds. This graph concisely represents the physical route environment in a discretized manner with information such as locations of the metering points and distances. For each pair $(\nu_e, \nu_r) \in \mathcal{N}_e \times \mathcal{N}_r$ there exists one route $(\nu_e, \nu_1, \ldots, \nu_m, \nu_r)$ in the graph $\mathcal{G}$ connecting the entry point ν_e to the runway threshold ν_r. Each aircraft follows exactly one of these routes corresponding to its entry point and its assigned runway. Fig.3.1 displays a model example of a route network: the Paris Charles de Gaulle (CDG) airport where we have four entry points $\mathcal{N}_e$ = {MOPAR, LORNI, OKIPA, BANOX} and two runways $\mathcal{N}_r$ = {27R, 26L}. $\mathcal{N}_w$ is composed of 10 nodes and $\mathcal{L}$ of 18 links. In this example, each TMA entry point corresponds to two routes specified by different runway assignments.

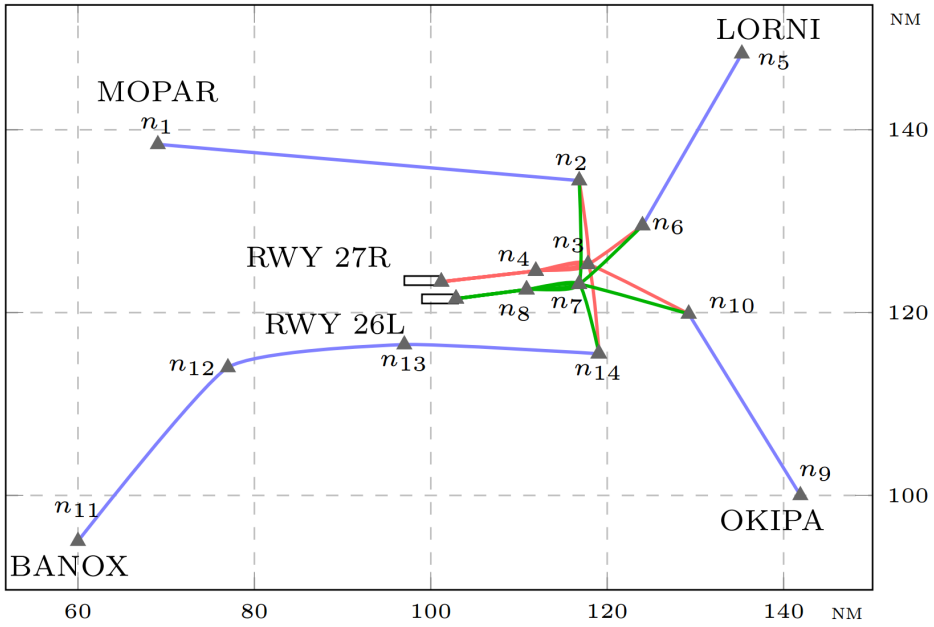


Figure 3.1: Arrival route structure of the Paris CDG Airport in west configuration represented by network abstraction.

3.1.3 Given data

Assume that we are given a set of landing flights (or aircraft) $\mathcal{F}$, for $f \in \mathcal{F}$ the following data are given:

- $e_f \in \mathcal{N}_e$: Entry point at the TMA;
- t_f^o : TMA entry time (Required flight arrival time at e_f);
- v_f^o : Entering speed at the TMA;
- c_f : Wake turbulence category;
- r_f^o : Initial assigned landing runway.

When we simulate the flight trajectory in the TMA, a possible speed profile should be provided. By referring to the real ground speed profile from the radar data, a continuously decelerated speed profile is implemented. In this case, the speeds of flights are decreasing with a constant deceleration rate until the Final Arrival Fix (FAF), the final speeds are defined as 110 kt, 130 kt, and 150 kt for aircraft with the categories heavy, medium and small, respectively.

3.1.4 Optimization models

The optimization model is established based on the deterministic model proposed in Ma *et al.* (2019). The conflict detection method is modified to adapt to the presence of uncertainty.

Decision variables

To be in line with the flight operational environment during approach, we propose three kinds of decision variables for each flight $f \in \mathcal{F}$ with specific decision ranges:

- t_f: the assigned TMA entry time of flight f, is adjusted by the number of time slots Δt (typically the slot $\Delta t = 5s$). In order to shift the TMA entry time, we can either delay it or speed it up during the en-route phase. The maximum earliness and tardiness with respect to the initial TMA entry time is denoted as $t_f^{\min}$ and $t_f^{\max}$, respectively. In practice, delay is always more frequent than the earliness due to perturbations such as weather, safety margin, human factors etc., thus we have $|t_f^{\max}| \geq |t_f^{\min}|$. In this study, $t_f^{\max}$ and $t_f^{\min}$ are set to be 20 mins and -5 mins, respectively. More formally, we have:

$$t_f \in T_f := \{t_f^o + j\Delta t | j \in \mathbb{Z} \text{ and } t_f^{\min} \leq j\Delta t \leq t_f^{\max}\}$$

- v_f: the assigned TMA entry speed of flight f, is selected by involving a user-defined speed increment Δ_f^v with respect to the initial TMA entry speed. $v_f^{\min}$ and $v_f^{\max}$ are given input data corresponding to the minimum and maximum allowable speeds for flight f. In this study, we set $v_f^{\min} = 0.9v_f^o$, $v_f^{\max} = 1.1v_f^o$ and $\Delta_f^v = 0.01v_f^o$. The TMA entry speed is given as:

$$v_f \in V_f := \{v_f^o + j\Delta_f^v | j \in \mathbb{Z} \text{ and } v_f^{\min} \leq j\Delta_f^v \leq v_f^{\max}\}$$

- r_f: the assigned runway to flight f, which is chosen from the runway set:

$$r_f \in \mathcal{N}_r$$

The decision vector associated with an arbitrary flight f is denoted as $\mathbf{x}_f = (t_f, v_f, r_f)$ and the whole decision vector is denoted as $\mathbf{x}$. By assumption, there exists a unique route $R_f(\mathbf{x})$ connecting the entry point e_f and the runway threshold of r_f. Moreover, using the speed profile of f we can predict the arrival time at each node of the route $R_f(\mathbf{x})$. Let us denote the arrival time corresponding to the i-th node along its arrival route of $R_f(\mathbf{x})$ as $t_{f,i}(\mathbf{x}_f)$, or simply $t_{f,i}$ in the sequel. Depending on the context, if no sequential requirement is put forward, we will also explicitly indicate the concerned node as ν, where the arrival time will be represented as $t_{f,\nu}(\mathbf{x}_f)$ or simply $t_{f,\nu}$.

Separation rules and conflict detection

To ensure flight safety, three separation requirements are considered: wake turbulence constraints, horizontal separation, and runway separation.

- Wake turbulence separation: The separation between two successively operated aircraft f and g should satisfy the wake turbulence separation minima $s_{f,g}$ associated with their aircraft categories c_f, c_g. The minimum separation standards are given in Table. 3.1.

- Horizontal separation: Aircraft must satisfy a minimum horizontal separation $h = 3NM$ based on the radar separation in the TMA when they are not flying on the same link.

- Runway separation: The runway separation requirements specified for two aircraft landing at the same runway with respect to their associated aircraft categories are given in Table. 3.2 (Frankovich, 2012). These requirements are also related to the associated occupancy times of aircraft on the runway.

A violation of the aforementioned separation rules gives rise to three types of conflicts. A *link conflict* refers to a violation of minimum wake turbulence separation between two successive aircraft flying on the same link. Catch-up conflict and overtake conflicts are examples of such conflict. A *node conflict* refers to the loss of horizontal separation between two aircraft on two different links adjacent to the same node. Finally, a *runway conflict* refers to the loss of runway separation between two aircraft landing at the same runway.

Table 3.1: Wake turbulence separation minima for different categories of aircraft $s_{f,g}$, with associated aircraft categories, in NM.

Categories		Leading Aircraft f		
		Heavy	Medium	Light
Trailing Aircraft, g	Heavy	4	3	3
	Medium	5	3	3
	Light	6	5	3

In order to adapt to a time-based metering system, which is convenient for conflict detection and metering the flows, the following lemmas transfer the distance-based separation to a time-based separation. Their proofs are presented later in Appendix A. Let (f, g) be a pair of distinct aircraft operated consecutively.

Lemma 1 *Let ν be a node belonging to the routes of f and g. Assume that aircraft f passes node ν before g. Then, there is no horizontal conflict between f and g at node ν if and only if:*

$$t_{g,\nu} - t_{f,\nu} \geq \tau^{nc}_{f,g,\nu} := \begin{cases} a, & \text{if } f \text{ and } g \text{ enter } \nu \text{ through the same link (see Fig. 3.2a),} \\ \max(a,b), & \text{if } f \text{ and } g \text{ enter } \nu \text{ from different links (see Fig. 3.2b).} \end{cases} \tag{3.1}$$

where

$$a = \begin{cases} \frac{h}{v_{g,\nu}}, & \text{if } v_{g,\nu} \leq v_{f,\nu}\cos\theta, \\ \frac{h}{v_{f,\nu}}, & \text{if } v_{f,\nu} \leq v_{g,\nu}\cos\theta, \\ \frac{h\sqrt{(v^2_{f,\nu}+v^2_{g,\nu}-2v_{f,\nu}v_{g,\nu}\cos\theta)}}{v_{g,\nu}v_{f,\nu}\sin\theta}, & \text{otherwise} \end{cases} \tag{3.2}$$

$$b = \begin{cases} \frac{h}{v_{g,\nu}}, & \text{if } v_{f,\nu}\cos\varphi \leq v_{g,\nu}, \\ \frac{h\sqrt{(v^2_{f,\nu}+v^2_{g,\nu}-2v_{f,\nu}v_{g,\nu}\cos\varphi)}}{v_{g,\nu}v_{f,\nu}\sin\varphi}, & \text{otherwise} \end{cases}$$

The angles θ and φ are shown in Fig. 3.2. Moreover, $v_{f,\nu}$, $v_{g,\nu}$ denote the speeds of aircraft f and g, respectively, at node ν. We remind that h represents the horizontal separation requirement which equals to 3NM.

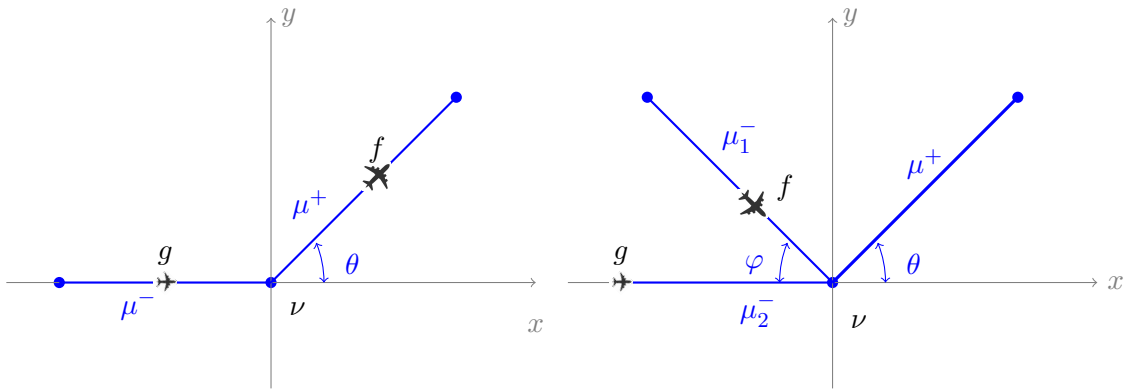
Lemma 2 *Let $\mu = (\nu, \omega)$ be a link belonging to the routes of f and g. Assume that aircraft f enters the link μ before g as shown in Fig. 3.3. The average speeds on μ for both aircraft are computed and denoted as $v_{f,\mu}$ and $v_{g,\mu}$. Then, there is no conflict between f and g on link μ if and only if:*

$$t_{g,\nu} - t_{f,\nu} \geq \tau^{lc}_{f,g,\mu} := \max\left(\frac{s_{f,g}}{v_{f,\mu}}, \frac{s_{f,g}}{v_{g,\mu}} + \frac{l(v_{g,\mu} - v_{f,\mu})}{v_{g,\mu}v_{f,\mu}}\right) \tag{3.3}$$

where l represents the length of the link μ. Remind that $t_{f,\nu}$ and $t_{g,\nu}$ represent the times that aircraft f and g enter the link μ, respectively. $s_{f,g}$ is the wake turbulence separation requirement associated with the categories of aircraft f and g (see Tab. 3.1).

Table 3.2: Single runway separation requirements, $\tau^{rc}_{f,g}$, with associated aircraft categories, in seconds.

Categories		Leading Aircraft f		
		Heavy	Medium	Light
Trailing Aircraft, g	Heavy	96	60	60
	Medium	157	69	69
	Light	207	123	82



(a) Aircraft f and g arrive at node ν through the same link.

(b) Aircraft f and g converge to node ν from different links.

Figure 3.2: Node conflict detection scenarios regarding different trajectory interaction patterns of flight pair.

Lemma 3 *If f and g land at the same runway r and f lands before g, then there is no runway conflict between f and g if and only if:*

$$t_{g,r} - t_{f,r} \geq \tau_{f,g}^{rc} \tag{3.4}$$

where $\tau_{f,g}^{rc}$ is given in Tab. 3.2 determined by the associated aircraft categories of aircraft f and g. $t_{g,r}$ and $t_{f,r}$ represent the arrival times of aircraft g and f at the runway threshold.

Based on the above mentioned lemmas, we introduce the following conflict indicators for each pair of aircraft (f, g).

- Node conflict indicator

$$\phi_{f,g,\nu}^{\text{nc}} = \begin{cases} 1, & \text{if } 0 \leq t_{g,v} - t_{f,\nu} \leq \tau_{f,g,\nu}^{nc} \text{ or } 0 \leq t_{f,\nu} - t_{g,\nu} \leq \tau_{g,f,\nu}^{nc}, \forall \nu \in R_f(\mathbf{x}) \cap R_g(\mathbf{x}) \cap \mathcal{N}_w \\ 0, & \text{otherwise} \end{cases} \tag{3.5}$$

- Link conflict indicator,

$$\phi_{f,g,\mu}^{\text{lc}} = \begin{cases} 1, & \text{if } 0 \leq t_{g,\nu} - t_{f,\nu} \leq \tau_{f,g,\mu}^{lc} \text{ or } 0 \leq t_{f,\nu} - t_{g,\nu} \leq \tau_{g,f,\mu}^{lc}, \forall \mu \in R_f(\mathbf{x}) \cap R_g(\mathbf{x}) \cap \mathcal{L}, \\ 0, & \text{otherwise} \end{cases} \quad \mu = (\nu, \omega), \tag{3.6}$$

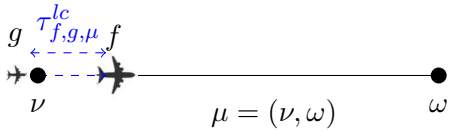


Figure 3.3: Link conflict detection configuration for two aircraft entering the same link consecutively.

- Runway conflict indicator

$$\phi^{\text{rc}}_{f,g} = \begin{cases} 1, & \text{if } 0 \le t_{g,r} - t_{f,r} \le \tau^{rc}_{f,g} \text{ or } 0 \le t_{f,r} - t_{g,r} \le \tau^{rc}_{g,f}, \\ 0, & \text{otherwise} \end{cases} \qquad \text{if } r_f = r_g = r \in \mathcal{N}_r, \tag{3.7}$$

The abusing notation $R_f(\mathbf{x}) \cap R_g(\mathbf{x}) \cap \mathcal{N}_w$ (resp. $R_f(\mathbf{x}) \cap R_g(\mathbf{x}) \cap \mathcal{L}$) designates the set of common nodes (resp. links) between the routes of f and g. It should be noted that when aircraft are diverged to different landing runways, vertical separation will be imposed. For example, in Fig. 3.1, the arrival routes in red guide the aircraft to runway 27R and the green ones show the routes for aircraft landing at 26L. The altitudes assigned for the aircraft landing at different runways satisfy the vertical separation requirement and the horizontal separation for aircraft with different altitudes are not considered.

Objective function of the deterministic model

The objective function of this deterministic model (DERM) minimizes the total number of link conflicts, node conflicts, and runway conflicts in the network.

$$G_d(\mathbf{x}) := \sum_{\substack{f,g \in \mathcal{F}, \\ g>f}} \sum_{\nu \in R_f(\mathbf{x}) \cap R_g(\mathbf{x}) \cap \mathcal{N}_w} \phi^{\text{nc}}_{f,g,\nu} + \sum_{\substack{f,g \in \mathcal{F}, \\ g>f}} \sum_{\mu \in R_f(\mathbf{x}) \cap R_g(\mathbf{x}) \cap \mathcal{L}} \phi^{\text{lc}}_{f,g,\mu} + \sum_{\substack{f,g \in \mathcal{F}, \\ g>f | r_f = r_g}} \phi^{\text{rc}}_{f,g} \tag{3.8}$$

where the first term in the sum counts the number of node conflicts, the second term counts the number of link conflicts and the last one counts the number of runway conflicts. Condition $g > f$ prevents double counting of conflicts.

Objective function for the probabilistic-based model under the presence of uncertainty

To incorporate the impact of uncertainty into the model, one typical way is to consider the exact information with randomness. Based on DERM, the probabilistic model (PROM) needs to bridge the conflict detection with the potential prediction error. A natural way is to consider the arrival time information of the flights at particular waypoints as random variables.

Let us replace the deterministic arrival time at node ν: $t_{f,\nu}$ by a random variable $T_{f,\nu}$ with a probability density function $\psi_{f,\nu}$. In the same way, conflict indicators (3.5,3.6,3.7) become random variables indicating the occurrence of a conflict and are represented as $\Phi^{\text{nc}}_{f,g,\nu}$, $\Phi^{\text{lc}}_{f,g,\mu}$, $\Phi^{\text{rc}}_{f,g}$, respectively. By replacing the relative terms in (3.8), the objective

function can be updated as:

$$
\begin{aligned}
G_p(\mathbf{x}) &= \sum_{\substack{f,g\in\mathcal{F},\\ g>f}} \sum_{\nu\in R_f(\mathbf{x})\cap R_g(\mathbf{x})\cap\mathcal{N}_w} P(\Phi^{\mathrm{nc}}_{f,g,\nu}=1) + \sum_{\substack{f,g\in\mathcal{F},\\ g>f}} \sum_{\mu\in R_f(\mathbf{x})\cap R_g(\mathbf{x})\cap\mathcal{L}} P(\Phi^{\mathrm{lc}}_{f,g,\mu}=1) + \sum_{\substack{f,g\in\mathcal{F},\\ g>f|r_f=r_g}} P(\Phi^{\mathrm{rc}}_{f,g}=1) \\
&= \sum_{\substack{f,g\in\mathcal{F},\\ g>f}} \sum_{\nu\in R_f(\mathbf{x})\cap R_g(\mathbf{x})\cap\mathcal{N}_w} E(\Phi^{\mathrm{nc}}_{f,g,\nu}) + \sum_{\substack{f,g\in\mathcal{F},\\ g>f}} \sum_{\mu\in R_f(\mathbf{x})\cap R_g(\mathbf{x})\cap\mathcal{L}} E(\Phi^{\mathrm{lc}}_{f,g,\mu}) + \sum_{\substack{f,g\in\mathcal{F},\\ g>f|r_f=r_g}} E(\Phi^{\mathrm{rc}}_{f,g}) \\
&= E\Big(\sum_{\substack{f,g\in\mathcal{F},\\ g>f}} \sum_{\nu\in R_f(\mathbf{x})\cap R_g(\mathbf{x})\cap\mathcal{N}_w} \Phi^{\mathrm{nc}}_{f,g,\nu}\Big) + E\Big(\sum_{\substack{f,g\in\mathcal{F},\\ g>f}} \sum_{\substack{\mu\in R_f(\mathbf{x})\cap R_g(\mathbf{x})\cap\mathcal{L}\\ \mu=(\nu,\omega)}} \Phi^{\mathrm{nc}}_{f,g,\mu}\Big) + E\Big(\sum_{\substack{f,g\in\mathcal{F},\\ g>f|r_f=r_g}} \Phi^{\mathrm{rc}}_{f,g}\Big)
\end{aligned}
\tag{3.9}
$$

where $E(\Phi)$ represents the expected value of the associated conflict indicator Φ. Respectively, the three terms represent the expected number of conflicts on nodes, links, and runways.

In order to compute the objective (Eq. 3.9), the airborne uncertainty model must be specified. For the uncertainty model, there is no restriction in selecting the distribution for the random variable. A natural choice is a Gaussian distribution, which has been widely used (Gonze *et al.*, 2018; Meyn and Erzberger, 2005; Caron *et al.*, 2013; Khassiba *et al.*, 2020). We can also use more complex models such as a mixture of two distributions, where one of the distributions can be used to model the uncertainty for regular flights and the other one to model flights with heavy delays. For the airborne uncertainty model, the chosen distribution must satisfy the flight constraint of Eq. 4.4 in the work of Marceau Caron (2014), which implies that the support of the density function is bounded (where the support of the density function refers to the range of real numbers for which the density function is positive). Suitable candidates for such properties are triangular and beta probability density functions. One example is the Program Evaluation and Review Technique (PERT), which is used in project management tools for characterizing the length of a task in a scheduling problem. The PERT distribution is defined as a Beta distribution with an assumption on the expected value regarding the minimum, most likely and maximum values of the variable and therefore provides a control on the probability assignment (Cottrell, 1999). Indeed, the accurate prediction of uncertainties along an entire aircraft's trajectory is not trivial. It is a function of space and time, which requires a mathematical expression for the uncertainty propagation mechanism along the route.

In our study, normal distribution is implemented for three reasons. First, the underlying computational characteristics of normal distribution are adapted to derive an analytical expression for the objective function of the proposed model. Second, in the absence of data to estimate the underlying distribution and by virtue of the central limit theorem, this choice seems reasonable. Third, the characteristics of normal distribution enable a unified expression of uncertain parameters, thus providing an analytical expression for conflict assessment. In fact, many other kinds of error distributions can be used, however, when no explicit representation of uncertainty propagation is available, the computational cost is expensive when exploring the scenarios using the repeated sampling method. Therefore, in this study, we assume that the prediction errors of aircraft arrival time at each waypoint ν (node) is approximately normally distributed, for which the variance is proposed to be proportional to the look-ahead time (the difference between the predicted and the current time, which corresponds to the time taken to make decisions). In our assumption, this

means that the uncertainty is supposed to grow with respect to a reference time (current time). More precisely, for an arbitrary flight f, the random variable corresponding to the TMA entry time can be represented as:

$$T_{f,0} \sim \mathcal{N}(t_{f,0}, \alpha(t_{f,0} - t_c)) \tag{3.10}$$

where t_c denotes the current time, $t_{f,0}$ denotes the predicted arrival time at the TMA entry node, α is a coefficient that defines the propagation rate of uncertainty, which is reflected in the variance of the probability distribution related to the look-ahead time. The uncertainties defined for the rest of the waypoints of flight f are formulated based on Markov assumption, that is to say, the probability of the arrival time at the next point $(T_{f,i+1})$ is independent on the past arrival time $(T_{f,0}, \ldots, T_{f,i-1})$, while it is conditionally related to the last arrival time $T_{f,i}$. We recall that in this notation, $T_{f,i}$ refers to the random variable representing the arrival time of the i-th waypoint on the arrival route of flight f. Thus, the following relation is established:

$$T_{f,i+1} | (T_{f,i} = t_i) \sim \mathcal{N}\left(t_i + (t_{f,i+1} - t_{f,i}), \alpha(t_{f,i+1} - t_{f,i})\right) \tag{3.11}$$

From this we can deduce that:

$$T_{f,i} \sim \mathcal{N}(t_{f,i}, \alpha(t_{f,i} - t_c)) \tag{3.12}$$

Then, for the current setting of the uncertainty model, the components of Eq. 3.9 can be calculated by referring to the characteristics of normal distribution. For two aircraft f and g that overfly the same node ν consecutively, $T_{f,v}$ and $T_{g,\nu}$ are independent random variables, and the probability density function of the inter-arrival random variable $T_{g,\nu} - T_{f,\nu}$ is given by $\psi_{g,\nu} * \psi_{f,\nu}$, where $*$ is the convolution operator. Thus, we can infer that the inter-arrival time is also normally distributed:

$$T_{g,\nu} - T_{f,\nu} \sim \mathcal{N}\left(t_{g,\nu} - t_{f,\nu}, \alpha(t_{g,\nu} + t_{f,\nu} - 2t_c)\right) \tag{3.13}$$

Therefore, an analytical computation of the probabilities of the conflict that constitute the objective function (Eq. 3.9) can be conducted according the following steps:

$$\begin{aligned} P(\Phi^{\mathrm{nc}}_{f,g,\nu} = 1) &= P(0 \leq T_{g,\nu} - T_{f,\nu} \leq \tau^{nc}_{f,g,\nu}) + P(0 \leq T_{f,\nu} - T_{g,\nu} \leq \tau^{nc}_{g,f,\nu}) \\ &= F_{f,g,\nu}(\tau^{nc}_{f,g,\nu}) - F_{f,g,\nu}(0) + F_{g,f,\nu}(\tau^{nc}_{g,f,\nu}) - F_{g,f,\nu}(0) \end{aligned} \tag{3.14}$$

where $F_{f,g,\nu}$ is the cumulative distribution function of the normal distribution associated with the inter-arrival random variable. Here we take the first term of Eq. 3.9 as an example to show the computational process. Similarly, other terms which are related to link conflicts and runway conflicts can be computed as well. Since the parameter α affects the variance of the random variable, it should be determined based on the time prediction error of airborne flights. Moreover, in the optimization, the value of α implicitly affects the overall separation assignment for the flight set. If α is set with a large value, an enlarged separation is required for each pair of flights to achieve small objective values. Empirically, α is set to 1.

Additional separation buffer based model

Considering the model formulation, PROM is naturally expected to perform better than DERM when uncertainty arises. Therefore the approach that applies additional buffers proposed in the previous work of Scala *et al.* (2019), is presented in our study and used as a comparative model. This model is denoted as BUFM and is established on the basis of DERM by enlarging wake turbulence separation requirements, horizontal separation requirement and runway separation requirements with a given percentage κ. The value of κ is chosen to be 20% which is a middle value used in Scala *et al.* (2019). It is worth mentioning that the conflict detection model used for BUFM is the same as the one used for DERM.

3.2 Solution approach

For problems related to arrival management, computational efficiency is essential. The heuristic algorithms are more likely to provide a fast and satisfying result, in which Simulated Annealing (SA) has been proven to be efficient and easy to be adapted to a complex problem that involves large-dimensional state space. Therefore, a time decomposition and sliding window approach combined with the SA algorithm that is similar to the one used in Ma *et al.* (2019) is adapted for this study.

3.2.1 Implementation of SA

SA has been thoroughly introduced in Chapter 2. In order to efficiently solve this problem, two specific procedures in SA for solving this problem are introduced.

- **Neighborhood Selection:** The neighborhood selection design for this problem is adjusted based on the one proposed in Ma *et al.* (2019). In the original neighborhood function, the number of conflicts encountered by a flight is recorded as its cost. In order to increase the chance of selecting high-cost aircraft, the roulette wheel selection method is used, in which the cost of each flight is proportional to its likelihood of being selected. The chosen flight from this approach will undergo a decision variable change. The advantage of this design is that the flights with higher costs are easier to be selected and the decision changes for those flights are more likely to be beneficial in minimizing the objective value. However, under the current design, only the flights whose costs are bigger than 0 can be selected. Considering the fact that all flights involved in optimization have potential effects on the overall performance of the model, we introduce a small value $\varsigma = 0.0001$ to be added to all flights, regardless of their initial costs. Since ς is small, on the one hand, this change has no significant effect on the probability of a flight being selected. On the other hand, it allows the algorithm to exploit the solution space deeper by being able to change the decision variables of all flights. To summarize, a detailed description is shown in Algorithm 2. Note that, for the PROM, the cost associated with each flight is the sum of the probabilities of conflict that an aircraft encountered in the whole network.

Algorithm 2 Neighborhood function

Require: For each flight f, record its cost c_f. Generate a random number ϵ = random (0,1);

The total cost $C_t = \sum_{f \in \mathcal{F}} (c_f + \varsigma)$;
sum $\leftarrow 0$;
target $\leftarrow C_t \cdot \epsilon$;
$i \leftarrow$1;
while sum<target **do**
    sum $\leftarrow$ sum+ $c_f + \varsigma$;
    $i \leftarrow i + 1$;
end while
For i-th flight, choose one of the decision variables with equal probability, then randomly choose one value between the minimum and maximum allowed deviations of that variable.

Table 3.3: Empirically-set parameter values of the simulated annealing algorithm with time decomposition approach.

Parameters	Values
Geometrical temperature reduction coefficient	0.99
Number of transitions of each temperature step	100
Final temperature	$10^{-4}T_0$
Time length of the sliding window	2h
Time shift of the sliding window	1h

- **Objective evaluation:** The objective evaluation benefits from the prediction of aircraft trajectories based on the choice of decision variables. Each time a flight is chosen to change the value of one of its decision variables, its trajectory information is predicted by the simulation model based on the new decisions. By comparing the arrival time differences of a flight pair to the required time separations on specific resources, the number of conflicts (for the PROM, the sum of probabilities of conflict) can be obtained, and then we can compute the objective value.
- **Stopping criterion:** the algorithm stops when the temperature reaches a predefined final temperature or an optimal solution is found (Objective value equals to 0).

Parameters related to temperature, neighborhood selection, and solution acceptance are provided in Tab. 3.3. The parameters are mostly set through empirical analysis to ensure a suitable adaptation of the algorithm to our problem. The pseudo-code of the cooling procedure of the SA is given in Algorithm 1 represented in Chapter 2.

3.2.2 Time decomposition and sliding window approach

The time decomposition and sliding window approach is usually implemented with other optimization algorithms in order to divide the whole problem into several sub-problems with small sizes, therefore reducing the complexity of the overall problem. In this study, the time decomposition and sliding window frame also provides a realistic environment for

the uncertainty propagation model by updating the current time and limiting the look-ahead time in a reasonable range. Several parameters need to be introduced and specified for this approach.

- T_I: starting time of the overall time horizon.
- T_E: ending time of the overall time horizon.
- L: time length of each sliding window.
- Q: shifting time of the sliding window.
- k: sequence number of the sliding window.
- S_k: starting time of the k-th sliding window, where $S_k = T_I + Q(k-1)$.
- E_k: ending time of the k-th sliding window, where $E_k = T_I + L + Q(k-1)$ and $E_k \leq T_E$.

As the name indicates, time windows are continuously sliding while solving the sub-problems. The window starts from an initial time T_I and slides until the end of the window arrives at the ending time of the time horizon T_E.

Flight status classification

In order to determine the flights that are involved in a given time window, a classification rule for flight status is proposed. For each flight, the status is assigned based on its earliest possible time for entering the TMA and the latest possible time of landing on the runway. The maximum transition time Γ_f of flight f in the TMA is deterministically predicted by considering the lowest TMA entry speed and the longer arrival route. Necessary parameters for determining the flight status are:

- τ_s^f: the earliest possible TMA entry time of flight f, which is computed by $t_f^o - t_f^{\min}$.
- τ_e^f: the latest possible landing time of flight f, which is computed by $t_f^o + t_f^{\max} + \Gamma_f$.

Suppose that we optimize the k-th window, the flights will be classified and marked with a status according to the following rule:

- **Completed flight:** $\tau_e^f < S_k$, this status means that the flight has been optimized in a previous window and has no interaction with the flights that need to be optimized in the current window.
- **Ongoing flight:** $\tau_s^f < S_k \bigwedge \tau_e^f > S_k$, here the notation of $\bigwedge$ indicates that the conditions on both sides of it should be satisfied. an ongoing flight means it has been optimized in the previous window, but it is still partially active in the network and may interact with flights that need to be optimized. Flights of this status are also included in the trajectory prediction but their decision variables are fixed.
- **Active flight:** $S_k < \tau_s^f \leq E_k$, the flights that need to be optimized in the current window are active flights.

Algorithm 3 Sliding window frame incorporating the SA and the neighborhood selection design.

$k \longleftarrow 1$;
$S_k \longleftarrow T_I$;
$E_k \longleftarrow T_I + L$;
$T_C \longleftarrow S_k$;
Assign different statuses to the flights based on the current window and the classification rules;
$\mathcal{F}_{act} \longleftarrow$ active flights, $\mathcal{F}_{ong} \longleftarrow$ ongoing flights;
while $S_k < T_E$ **do**
    Call Algorithm 1 and apply the neighborhood selection design (Algorithm 2).
    $k \longleftarrow k + 1$
    $S_k \longleftarrow S_k + Q$;
    $E_k \longleftarrow S_k + L$;
    $t_c \longleftarrow S_k$;
    Update the flight status regarding current window and current flight information;
    Update $\mathcal{F}_{act}$ and $\mathcal{F}_{ong}$ according to newly updated flight status;
end while

- **Planned flight:** $\tau_s^f > E_k$, planned flights have their earliest TMA entry time located later than the ending time of the current window. As the window slides afterward, the planned flights will go through the statuses from active to ongoing and then completed.

For each window, only the flights with active or ongoing status will be involved in the sub-problem of the current window.

Uncertainty management under sliding window frame

The time decomposition and sliding window frame enables a practical realization of uncertainty propagation model by providing regularly shifted current time t_c and limited look-ahead time. The current time t_c is set to be the time window starting time. The length of the time window identifies the aircraft that will arrive in a reasonable range as it is not realistic to control arriving aircraft too far in advance, which puts a constraint on the look-ahead time to avoid a strong increase of prediction error. Fig. 3.4 conceptually illustrates how the random variables vary under the sliding window frame. In each of the figures, a coordinate is established, in which the blue stripe represents the time horizon of the current sliding window in the x-axis, the y-axis represents the possible values of the variance, and the green line associates the possible predicted times with values of variance for the corresponding random variables. For each flight, predicted TMA entry time (at the TMA entry node) and predicted landing time (at runway threshold) displayed by the black arrows are shown on the timeline. The red lines represent the associated variances. Three flights f_1, f_2, f_3 are displayed in the figure with their associated statuses of that window. As the probability distribution of a random variable is partially determined by the current time t_c, the probability distributions for the same flight on the same waypoint are different when the window slides to the next one.

The procedures of the sliding window frame considering the management of uncertainty

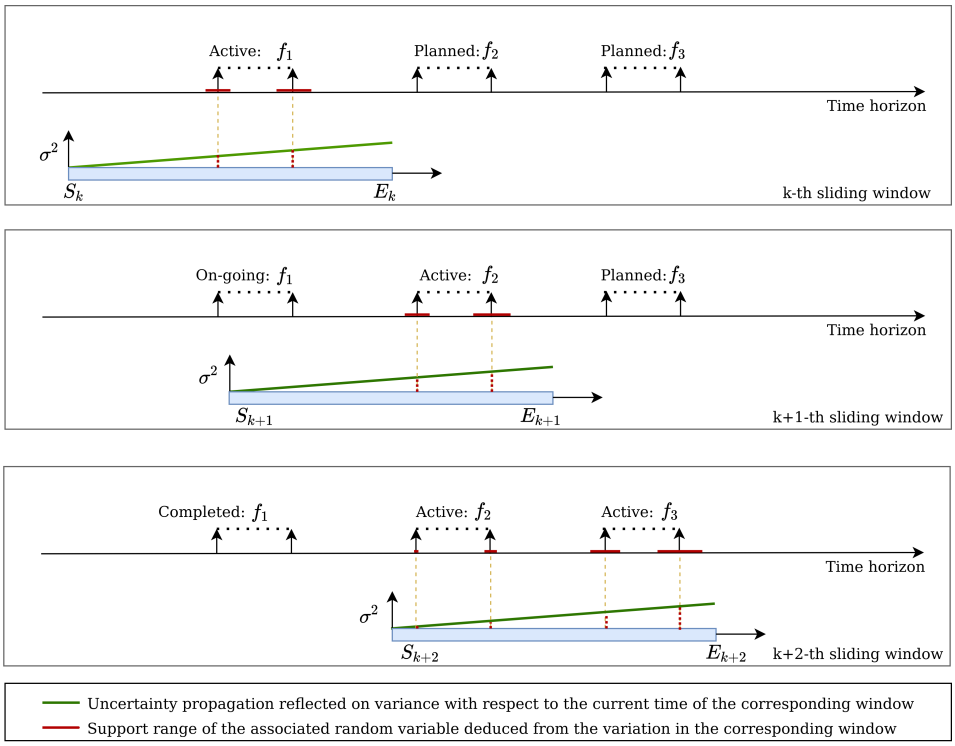


Figure 3.4: The uncertainty propagation for flights of different statuses under the sliding window frame.

are described in Algorithm 3. It is worth mentioning that though both active and ongoing flights are involved in the problem, only the active flights are optimized with decision changes.

3.3 Solutions evaluation

In this section, we describe a simulation environment to test the optimized solutions obtained from the DERM, BUFM and PROM by means of stochastic perturbation.

3.3.1 Simulation model for solution evaluation

The simulation model is established based on the Monte-Carlo approach. The main idea is to obtain a large number of outcomes by repeating the random perturbations on the optimized solutions so as to achieve the statistical analysis. We note that, in the process of evaluating the solution quality, the construction of a reliable simulation tool goes hand-in-hand with the development of the modeling frameworks for the optimization problem (Marla *et al.*, 2020). The practical issue considered in PROM in line with the real-life situation is that for each flight, the arrival time of the next waypoint always depends on the arrival time at the current one. The sampling process in the simulation has taken this fact into account, thus the arrival times at all the passed waypoints of a flight are perturbed. Through sampling, each experiment will lead to a deterministic evaluation. The number of conflicts generated from each experiment is calculated and considered as a metric of the robustness. According to the law of large numbers, the number of replications

is set to be 10000 to guarantee the accuracy of the statistical analysis.

3.3.2 Simulation scenarios

Considering how PROM and BUFM manage the uncertainty, two types of disturbance distribution are applied in the simulation model to evaluate the solution quality.

- Normally distributed perturbation: In this case, the probability distributions of the random variables, where the samples are drawn from, are consistent with the random variables in the PROM as indicated in Eq. 3.10 and Eq. 3.11 for the waypoints along the route of a flight.
- Uniformly distributed perturbation: In this case, the random variables where the samples are drawn from follow uniform distributions. Their variances are identically computed as in the previous case: $\alpha(t_{f,0}-t_c)$ on the TMA entry point and $\alpha(t_{f,i+1}-t_{f,i})$ from the waypoint i to the next one $i+1$. Mention that the variance of uniform distribution is defined as $(p-q)^2/12$ where p and q are the maximum and minimum bound of this uniform distribution. Thus, the probability distribution for the random variables whose ranges defines the sample space can be specified by Eq. 3.15 and Eq. 3.16 (in a form of $\mathcal{U}(p,q)$):

$$T_{f,0} \sim \mathcal{U}(t_{f,0} - \sqrt{3\alpha(t_0 - t_c)}, t_{f,0} + \sqrt{3\alpha(t_0 - t_c)}) \tag{3.15}$$

$$\begin{aligned} T_{f,i+1}|(T_{f,i} = t_i) \sim \mathcal{U}\Big(& t_i - t_{f,i} + t_{f,i+1} - \sqrt{3\alpha(t_{f,i+1} - t_{f,i})}, \\ & t_i - t_{f,i} + t_{f,i+1} + \sqrt{3\alpha(t_{f,i+1} - t_{f,i})}\Big) \end{aligned} \tag{3.16}$$

In order to investigate the solution performance under different levels of uncertainty, two values of α are applied in the simulation. Thereby four scenarios are presented with combinations of probability distributions and levels of uncertainty, namely:

- Scenario 1 (N1): samples are drawn from the domain of normally distributed random variables with a smaller variance by setting $\alpha = 1$.
- Scenario 2 (N2): samples are drawn from the domain of normally distributed random variables with a higher variance by setting $\alpha = 2$.
- Scenario 3 (U1): samples are drawn from the domain of uniformly distributed random variables with a smaller variance by setting $\alpha = 1$.
- Scenario 4 (U2): samples are drawn from the domain of uniformly distributed random variables with a higher variance by setting $\alpha = 2$.

3.3.3 Time decomposition and sliding window frame for simulation

In our simulation model for solution evaluation, the variance of the probability distribution associated with the flight arrival time at the TMA entry node is related to the current time t_c (Eq. 3.10). Since our data ranges from 0:00 to 24:00, it is clear that without the sliding

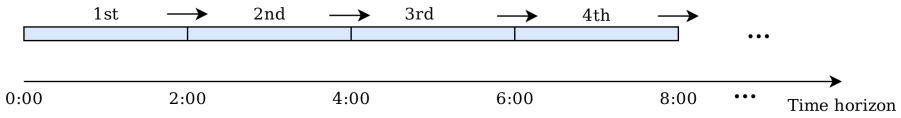


Figure 3.5: Sliding window frame applied in the simulation.

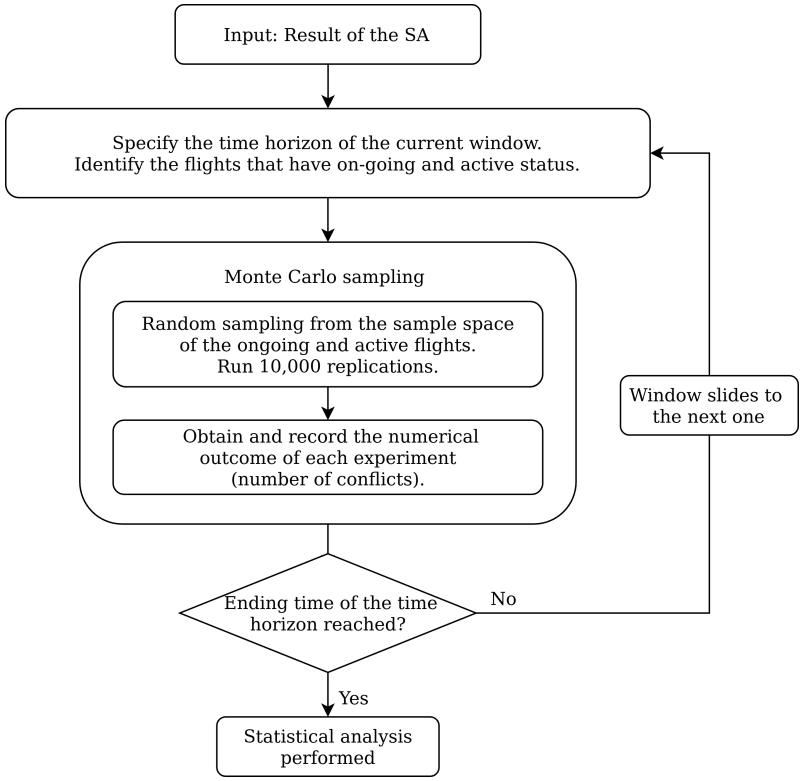


Figure 3.6: Overall process of the simulation framework for the solution evaluation.

window frame, the current time is fixed, hence the probability distributions of the random variables are unrealistic for some aircraft. Therefore, time decomposition and the sliding window frame is also applied in the simulation framework and the flight classification mechanism is applied for specifying the flights involved in the simulation for different time windows. As the schedules of all aircraft have been optimized, the most important evaluation lies on the solution performances regarding different traffic densities. In order to include all the aircraft in the simulation experiments and avoid excessive simulations, in the simulation model, we simply set the length of the sliding window L to 2 hours, and the whole day is divided into 12 windows with the shift time Q equals to 2 hours, as shown in Fig. 3.5. The simulations are then conducted for each time window separately, where the status of a flight is classified based on the exact information provided in the optimized solutions associated with this flight.

The overall process of the simulation framework for the solution evaluation taking the sliding window frame into account is illustrated in Fig. 3.6.

3.4 Case study and results

3.4.1 Case study

The actual arrival flight data on 18 February 2016 at the Paris CDG airport is applied to our model. There are two runway configurations at the Paris CDG airport: west-flow (26L, 27R — 26R, 27L) and east-flow (09L, 08R — 09R, 08L). This work focuses on the more frequently used west-flow configuration. The TMA route structure of the Paris CDG is displayed in Fig. 3.1.

The categories of operated flights are mostly large and medium. Tab. 3.4 provides the arrival flight information in the TMA. The overall optimization process is run on a 2.50GHz core i7 CPU, under a Linux operating system PC with Java code.

3.4.2 Results of one window sub-problem

Focusing on a single window allows a better comparison of the different models without the sliding effect. For example, in scenario N1, the sample space is consistent with the range of the random variables built in PROM model, therefore, we anticipate that the expected number of conflicts in the optimal solution of PROM is close to the average number of conflicts obtained from the simulation result of scenario N1.

In this single window case, the flights that are initially operated during the busy period from 6:00 to 8:00 are taken into account for our demonstration with respect to the optimization and solution evaluation processes. Associated information of arrival flights is given in Tab. 3.5. In this period, we have 78 flights and all of them are active flights.

Optimization results for the single window case

Considering the differences in the objective evaluations of the three models, the final results of the objective values have their own unique meanings. Tab. 3.6 displays the optimized objective values for the link, node and runway conflicts, for which the objective values

Table 3.4: Traffic flow distribution of the flight data on 18 February 2016 at the Paris CDG airport.

	Medium	Heavy	Total
MOPAR	73	40	113
LORNI	153	37	190
OKIPA	146	39	185
BANOX	82	10	92

Table 3.5: Information of the flights on 18th February 2016 between 6:00 to 8:00 at the Paris CDG airport.

	Medium	Heavy	Total
MOPAR	5	10	15
LORNI	21	3	24
OKIPA	21	2	23
BANOX	14	2	16

of BUFM and DERM are 0. The results of DERM suggest that all aircraft can be well managed with required separation minima. With the separation requirement increased by 20%, the objective function value of BUFM indicates that extra space between flight pairs is available under current traffic density, which further implies the potential of obtaining a robust solution. In contrast, the objective value of PROM that evaluates the expected number of conflicts is positive. This is because the uncertainty propagation model may yield random variables with big variances, therefore under limited time resources, the probability of conflict occurrence can not be eliminated.

The computational time of the optimization process is 30 seconds for PROM and 1.2 seconds for BUFM and DERM, respectively. Considering the window time horizon, they are all very efficient in solving the problems.

Simulation-based solution evaluation for the single window case

Once the optimization process is finished, the solutions are taken as inputs and randomly perturbed based on the predefined scenarios. The simulation results are displayed using statistical frequency distribution. Frequency here refers to the occurrence of a particular value of the conflict number in the network. Fig. 3.7 displays the comparison of frequency distributions for PROM, BUFM, DERM under four scenarios, in which we note that the distributions resemble a bell shape. Therefore, the statistical indicators of mean and standard deviation can provide a reference in the performance comparison. In the figure, no significant differences can be clearly observed for the standard deviation, while it is clear that PROM outperforms BUFM with less conflicts generated. For more detailed information, Tab. 3.7 aggregates the average conflicts arising in the simulation. By comparing the number of conflicts associated with the models in each scenario, we can see that PROM performs the best with the least number of conflicts generated.

It should be noted that the demonstration of the single window case is supposed to verify the connections between the PROM and the simulation model. As mentioned above, the final value of the objective function in PROM should be close to the simulation results obtained from scenarios N1. This can be observed by comparing the objective values in the last row of Tab. 3.6 and the simulation results in the fourth column in Tab. 3.7, where the differences are minor.

3.4.3 Results of optimization and solution evaluation for one day case

In the previous investigation, it has been proven that PROM provides a better solution than BUFM in a high traffic density case. We continue to investigate the model perfor-

Table 3.6: The final objective values of the optimization process for the window from 6:00-8:00

Objective value (Number/expected number of conflicts)	Node	link	Runway	Total
DERM	0	0	0	0
BUFM	0	0	0	0
PROM	16.67	28.32	11.79	56.78

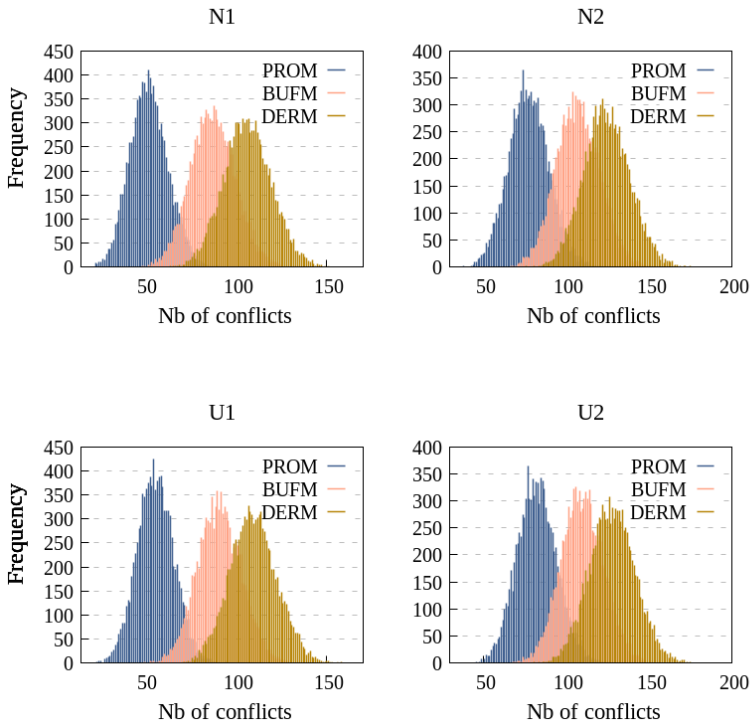


Figure 3.7: The frequency distributions of the simulation experiments in terms of number of conflicts. The simulation results are obtained from the Monte-Carlo simulation based on the optimized solutions of PROM, BUFM, DERM during the time windows from 6:00 to 8:00.

Table 3.7: Simulation results of average numbers of conflicts in a single window for each resource based on the optimized solutions of PROM, BUFM, and DERM under four scenarios.

	N1			**N2**		
Model	DERM	BUFM	PROM	DERM	BUFM	PROM
Node	33.19	25.04	13.9	38.64	30.35	21.66
Link	54.73	46.23	27.25	64.15	56.92	40.21
Runway	18.68	16.13	11.2	21.91	19.71	15.67
Total	**106.6**	**87.41**	**52.35**	**124.69**	**107.08**	**77.54**
	U1			**U2**		
Model	DERM	BUFM	PROM	DERM	BUFM	PROM
Node	34.13	26.07	14.85	39.62	31.08	23.34
Link	56.45	48.2	29.11	66	58.47	42.7
Runway	19.19	16.69	11.96	22.33	20.29	16.3
Total	**109.8**	**90.97**	**55.92**	**127.97**	**109.84**	**82.42**

mance in different problem sizes. Following the same steps as in the single window case, both the optimization results and the results for the solution evaluation are presented.

Table 3.8: Optimization results of PROM including the computational time for multiple problem sizes and the associated final objective values.

Time window	Active flights	On-going flights	Total	Execution time(s)	Objective value
0:00-2:00	2	0	2	0.03	0
1:00-3:00	1	2	3	0.008	0
2:00-4:00	3	1	4	0.026	0
3:00-:500	28	1	29	4.327	0.08
4:00-6:00	35	13	48	7.478	0.126
5:00-7:00	39	16	54	13.388	3.35
6:00-8:00	78	11	89	32.249	47.43
7:00-9:00	71	40	111	40.643	60.01
8:00-10:00	41	49	90	28.392	13.16
9:00-11:00	57	16	53	26.19	12.09
10:00-12:00	68	21	89	36.487	32.76
11:00-13:00	33	44	77	20.23	2.31
12:00-14:00	43	14	57	14.37	1.45
13:00-15:00	39	20	59	14.164	0.53
14:00-16:00	38	20	58	17.917	1.16
15:00-17:00	55	20	75	25.366	17.395
16:00-18:00	44	31	75	24.321	6.64
17:00-19:00	53	23	76	25.484	10.57
18:00-20:00	55	22	77	21.609	8.89
19:00-21:00	38	34	72	19.179	0.614
20:00-22:00	37	14	51	14.847	0.68
21:00-23:00	22	23	45	2.89	0
22:00-24:00	12	9	21	0.5	0

Optimization results of the one day case

Knowing that the air traffic densities in the sub-problems of the associated windows are different, in Tab. 3.8, the number of flights (ongoing and active) that define the sub-problem size and the associated final objective values are displayed. In addition, the execution times of associated sub-problems in the optimization process are also given for the 24 windows. As for DERM and BUFM, the final objective values are all 0 for all the sub-problems, which means conflict free solutions can be obtained for both DERM and BUFM. The total computational time of BUFM and DERM are 5 seconds and 2 seconds, respectively, which are both highly promising for solving a complex problem with a large volume of air traffic.

Results of solution evaluation for one day case

The realization of the sliding window framework leads to independent simulations of each window. In consequence, it is necessary to aggregate the results with a clear demonstration while displaying as much information as possible. Therefore, boxplot is selected to present the statistic features of the number of conflicts with a size of 10000 (number of replications) obtained from the simulation. In Fig. 3.8, four figures corresponding to the simulation

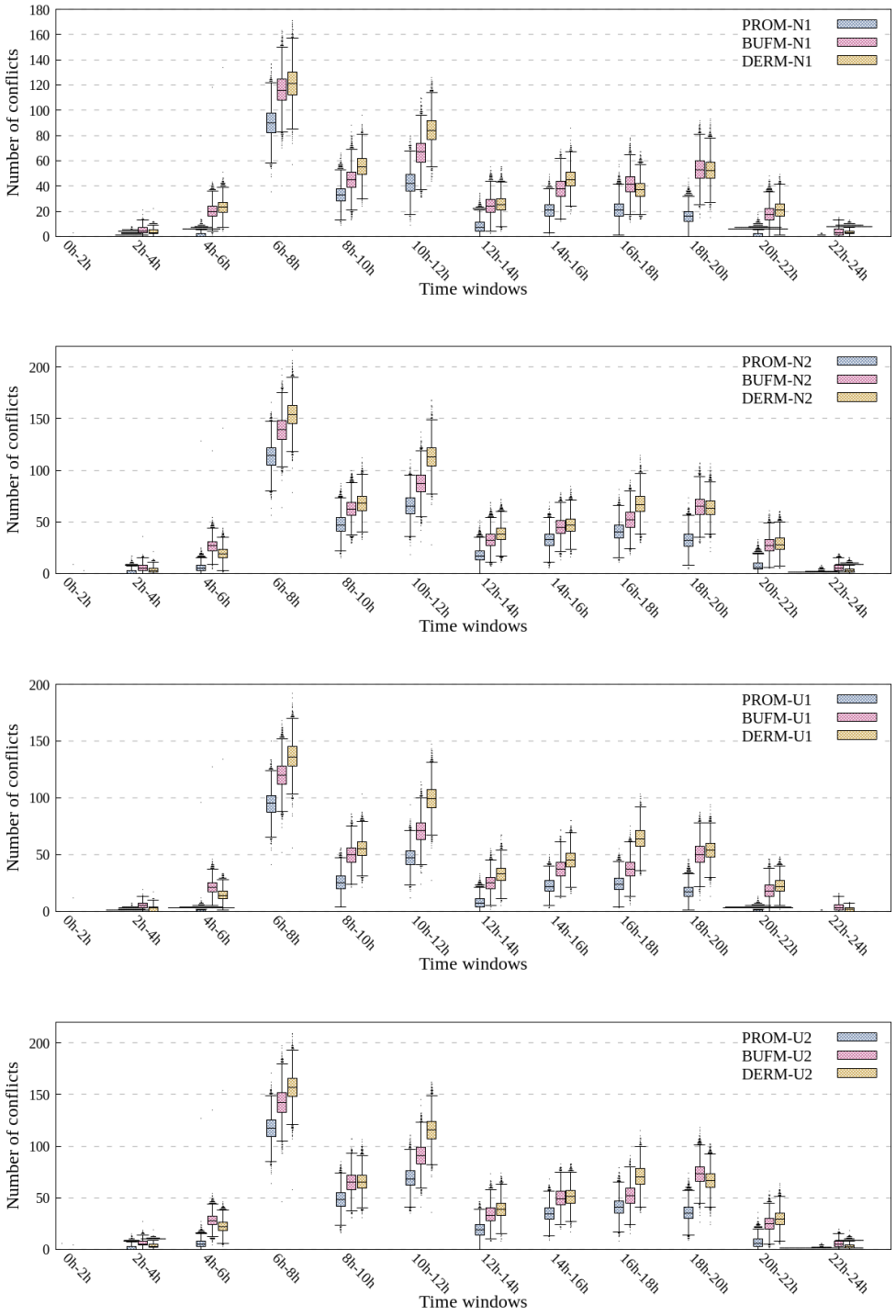


Figure 3.8: A comparison of simulation results of PROM, BUFM and DERM based on four scenarios presented by boxplot. The numbers of conflicts of each experiment conducted by perturbing aircraft are aggregated, from which we can observe the maximum value, minimum value, and median value associated with each set of data.

results for N1, N2, U1, and U2 from up to down are illustrated, in which the blue boxes represent the results of PROM, while the red and yellow boxes correspond to the results

associated with BUFM and DERM, respectively. x-axis indicates the time window, for which the simulations are conducted separately. Each boxplot contains various information related to the numbers of conflicts generated from the experiments, such as maximum value, minimum value, and median value. Moreover, we can also approximately observe the distribution range of the conflicts. Overall, the plots in the four figures show the same variation trend in terms of distribution of conflicts on the time windows. By checking the number of flights in the time windows, we can see that the number of active flights involved in the simulation has a significant impact on the number of conflicts generated for each model. We continue the comparison of the boxplot sets by focusing on the median values. It is obvious that PROM corresponds to the least number of conflicts compared to the results associated with BUFM and DERM for all of the time windows in the four scenarios. Comparing the results between BUFM and DERM, occasionally, BUFM has higher median values than DERM, which mostly appear in the case when there are not too many flights involved. This phenomenon suggests that the approach of inserting buffer is not practically reliable in reducing the potential risk caused by uncertainty.

In order to evaluate the quality of the solutions numerically, the solution performance indicator is evaluated as: $(C_{\mathrm{DERM}} - C_{\mathrm{X}})/C_{\mathrm{DERM}} \cdot 100$, where C_{DERM} is the total number of conflicts obtained from the simulation of DERM, and C_{X} can be either total number of conflicts associated with BUFM or PROM from the simulation. This indicator essentially reflects the robustness of the solutions obtained from the models that take uncertainty into account with respect to the deterministic model in terms of conflict absorption percentage. In Tab. 3.9, the performance indicator associated with PROM and BUFM are given. In the first time window, $*$ represents 0 conflict. The negative values are observed in some cases of BUFM, which is basically consistent with the cases shown in Fig. 3.8 where the median values of BUFM is higher than that of DERM.

We then investigate the variation of the performance indicators in Tab. 3.9 in contrast with the number of active flights. Considering the indicators associated with PROM (the 3rd, 5th, 7th and 9th row in Tab. 3.9), in all scenarios, high values appear in the time windows with only a small number of active flights (2nd and 12th time window) and their adjacent time windows (3rd, 11th time window). When the number of active flights increases, the performances slightly drop (5th, 6th, 7th, 8th, 9th, 10th time windows). For the busiest time window (4th time window), $25\% - 31\%$ of conflict absorption is achieved. However, the performance of BUFM shows no such pattern. When we focus on the perturbation with different levels of variance (N1 v.s N2 and U1 v.s U2), PROM shows a reduction in its performance in the scenario with a higher variance. However, BUFM shows no clear sign of how the performance indicators change. Moreover, except for the negative performance indicators, the highest performance indicator of BUFM is 28, which falls in the worst performance range of the results associated with PROM. To summarize, PROM generates a solution that performs best in the three models while hedging against all kinds of real-life perturbations. Its solution has also provided a certain level of stability in performance in all scenarios with all levels of traffic densities since positive performance indicators are always obtained.

The simulation model estimates the total number of conflicts composed of three types of conflicts. It is necessary to identify the origins of the conflicts so as to better characterize the effects of perturbation in terms of conflict distribution in the network. In Fig. 3.10,

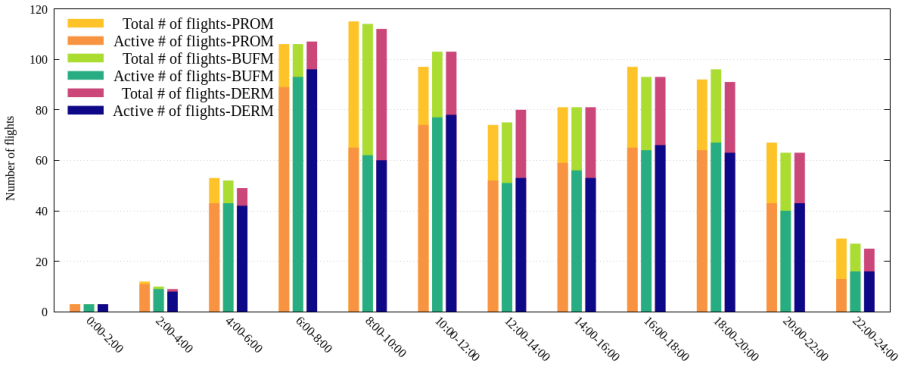
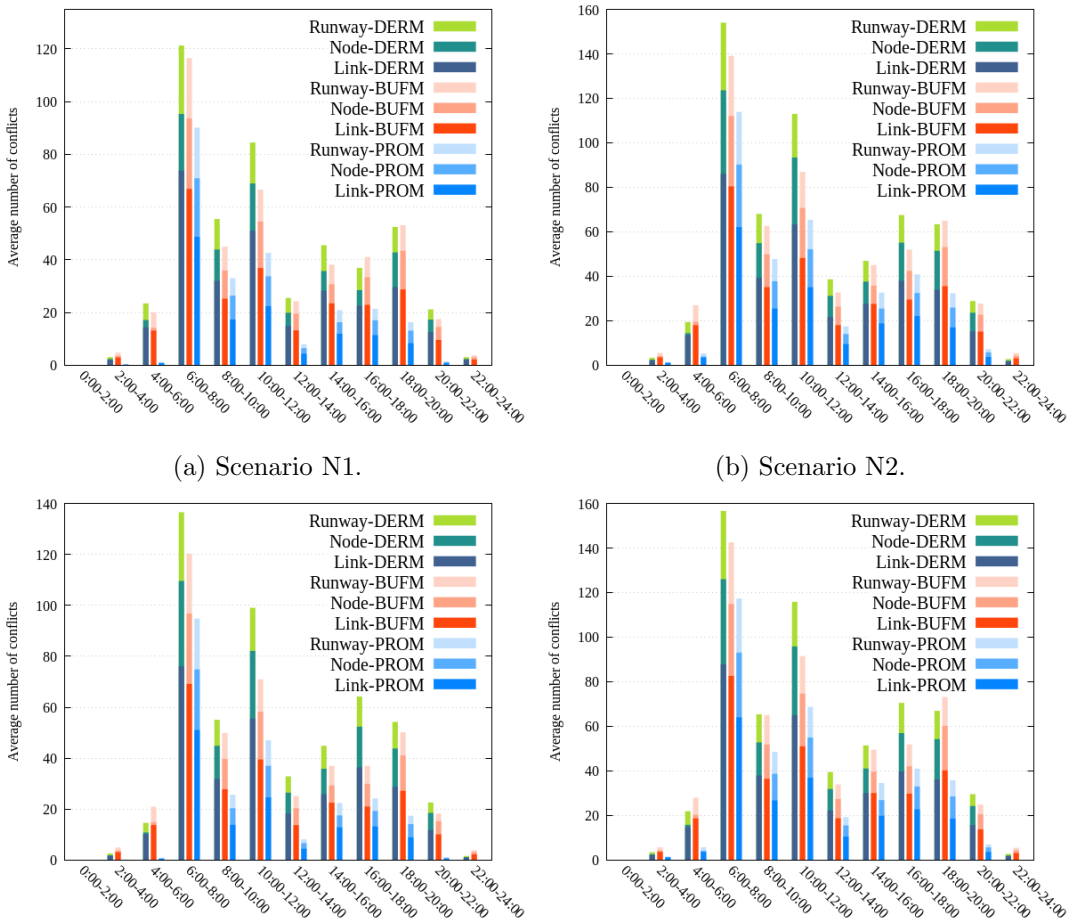


Figure 3.9: Number of active flights and total flights involved in each window in the simulation of the three models.

four subfigures associated with four scenarios are displayed. In each figure, the link, node and runway conflicts are represented by stacked bar graphs based on the average conflict number of that specific type. The average number of conflicts of each type shows similar variation trend to the total number of conflicts over time. Another observation is that the link conflicts account for the majority in the total conflicts. The reason for fewer node conflicts can be concluded from two key aspects. First, the conflict detection for a link and the entry node of this link are based on same time information, but in some cases, the time separation requirements computed for node conflicts are smaller than those of the link conflicts. (see Eq. 3.2 and Eq. 3.3). Second, the number of nodes considered in the conflict evaluation is less than the number of links. Considering the fact that node conflict detection is carried out for aircraft on different links. However, the operations of aircraft before entering the TMA are not concerned in our study, therefore, on the four node entry points, the node conflict evaluation is not conducted. In the network, the runway separation violations have the least chance of occurring, since only two runways are involved.

Table 3.9: Solution performances indicators of PROM and BUFM represented by the percentage of conflicts absorbed based on the results of DERM for 12 windows and four scenarios. The higher the performance indicators, the better the conflict absorption ability. In each case, the value of PROM is always higher than the value of BUFM.

	Time window	1	2	3	4	5	6	7	8	9	10	11	12
N1	BUFM (%)	*	-55	14	4	19	21	5	16	-11	-1	17	-22
	PROM (%)	*	90	95	26	40	50	69	54	42	69	93	100
N2	BUFM (%)	*	-64	-39	10	8	23	15	4	23	-3	4	-95
	PROM (%)	*	57	73	26	30	42	55	30	40	49	75	96
U1	BUFM (%)	*	-92	-43	12	9	28	24	18	42	7	19	-140
	PROM (%)	*	93	94	31	53	53	75	50	62	68	95	100
U2	BUFM (%)	*	-63	-28	9	1	21	14	4	26	-9	16	-98
	PROM (%)	*	53	74	25	26	41	51	33	42	47	76	98



(a) Scenario N1.

(b) Scenario N2.

(c) Scenario U1.

(d) Scenario U2.

Figure 3.10: Average number of conflicts comparison on links, nodes and runways of all the time windows based on four scenarios for the three models. The conflicts on links, nodes and runways of the same situation are represented by stack bar graph. The variation of each type of conflict is similar on time window under the same scenario. For each case, the number of link conflicts is higher than node conflicts, and the number of runway conflict accounts for the least of the total conflicts.

3.5 Conclusion

This study addresses the arrival aircraft scheduling problem under uncertainty by performing conflict detection and resolution for aircraft in a predefined network. A novel probabilistic model is proposed based on the deterministic model in Ma *et al.* (2019). The uncertainty is managed by representing the time information used for conflict detection as random variables. Considering the uncertainty propagation in reality, the probability distributions of the random variables are characterized based on the Markov assumption for a flight during the operation in the TMA. By assuming that the prediction error is normally distributed, we benefit from the properties of the normal distribution to produce an efficient way of evaluating the probability of conflict occurrence between each two successively operated aircraft at specific locations. The objective function is formulated as the expected number of conflicts of all flights in the network and it can be analyti-

cally computed. Using this design, the computational complexity is reduced compared to the typical uncertainty optimization approaches. In our study, uncertainty has been considered as an integrated term, therefore a wide application of this approach is possible.

The time decomposition and sliding window frame combined with simulated annealing has been proven to be efficient as a solution algorithm. For the proposed model, the sliding window frame also plays an important role in limiting the variance of the random variables to a reasonable range.

The work has been enriched by introducing a Monte-Carlo simulation framework with different scenarios representing the variability of the uncertainty elements. Except for the proposed model, DERM and BUFM represent the deterministic model without considering the uncertainty, and the model incorporating separation buffers are presented for the purpose of comparison. The solutions of the three models are evaluated by means of stochastic perturbation for 12 windows with different traffic densities. Statistical analyses are conducted by comparing the conflicts generated in different scenarios in the simulation. A relative comparison is conducted by using the simulation results associated with the deterministic model as the basis upon which the conflict absorption percentages for the other two models are computed. Numerical results show the limitation of BUFM, as in some cases, it performs even worse than DERM. In contrast, the proposed model provides a certain level of stability as its solution can still absorb 25% of the conflicts with respect to the deterministic model under a case of high traffic density and high level of perturbation. Moreover, the type of the distributions of the perturbation does not cause a notable difference in the performance of conflict absorption.

Compared to other uncertainty optimization approaches, the proposed method has advantages given different perspectives. For example, robust optimization attempts to seek an optimal solution under the worst-case scenario which leads to a conservative result. This conservatism will lead to a waste of available time slots in arrival management, thus reducing efficiency. However, the proposed approach does not apply hard constraints, so that the balance between the available resources and current traffic density can be achieved while improving robustness for the final result. Though belonging to the stochastic problem, in the proposed approach, the assumption of normality of the random variables eases the computational burden of using scenario-based solution approach (e.g. SAA). Therefore, it can have a broader application.

Chapter 4

Arrival air traffic optimization in the extended horizon

In this chapter, we address a dynamic/on-line problem for arrival air traffic considering a management range from the airspace close to the airport to enroute phase. The problem is established based on the concept of Extended Arrival MANagement (E-AMAN) proposed by Single European Sky ATM Research (SESAR). E-AMAN is one of the SESAR solutions that refers to extending the current Arrival MANagement (AMAN) horizon to further upstream so as to prepare earlier the sequencing of air traffic destined to a particular airport (SESAR, 2015*b*). The key benefits of this concept include reducing the holding time over the destination airport, improving control efficiency and lowering fuel costs and emissions.

This chapter is organized as follows: Section 4.1 describes the problem and introduces the details of the E-AMAN concept. Section 4.2 presents the mathematical model of the addressed problem. Then, in Section 4.3, the solution algorithm applied for this problem is presented. Further, the case study and the computational results are given in Section 4.4. Finally, conclusion is provided in Section 4.5.

4.1 Problem description

The current AMAN system mainly focuses on the arrival flights that are 100-200 NM away from the airport. Aircraft are operated based on known information and predefined plans without considering further situations in the Terminal Maneuvering Area (TMA). Standard procedures such as holding and stacking are commonly used by controllers in the TMA to make sure aircraft can obtain available landing slots. Though these procedures are useful, they might lead to undesirable results such as delay, fuel cost, noise effect, and extra workload for both controllers and pilots. E-AMAN is therefore proposed as a smart solution that attempts to alleviate the pre-mentioned issues by extending the AMAN horizon up to 500NM. With a longer management horizon, controllers in the upstream sectors are allowed to adjust aircraft trajectories earlier, achieving an early sequencing of air traffic and reducing the need of stacking and holding over the airport, thereby satisfying the interests of multiple stakeholders.

The deployment of E-AMAN also relies on the current available operational models and

systems. The arrival sequencing and scheduling manager for the TMA helps controllers to optimize capacity by giving associated instructions to pilots for conducting appropriate maneuvers taking into account safety, fuel consumption and efficiency in the terminal area and at runways. The enroute decision-making tool optimizes the service provision by assisting with detection and monitoring tasks. As an integration of the above-mentioned tools, E-AMAN requires all parties to share information using a System-Wide Information Management (SWIM) service (SESAR, 2019*b*). Based on the SWIM, the systems could provide useful advisories for controllers that are in charge of the higher (cruise) flight levels. Instructions are then delivered to the pilots who make adjustments to aircraft speed such that they meet the Time To Lose (TTL)/Time To Gain (TTG) before reaching the TMA.

Note that some domestic flights has relatively small operational ranges. Therefore, according to whether the flight horizon exceeds the limit of 500NM, flights are classified as long-haul flights and short-haul flights. Fig. 4.1 shows an application of E-AMAN, where the procedures and control manners for aircraft in different phases are displayed. The Controlled Time of Arrival (CTA), which is an imposed time constraint at a defined point associated with a route or a runway is introduced for aircraft. The CTA is computed by the AMAN system in accordance with the predicted information, aircraft operation limitations, safety requirements etc. Then the CTA is proposed to the pilots by the controllers, based on which flight on-board computer is used to achieve required accuracy (SESAR, 2015*a*).

Adapted to the E-AMAN concept, this study focuses on the arrival traffic within the boundary of 500NM from the destination airport. Aircraft that enter the boundary are operated in the enroute network, then they follow the predefined Standard Terminal Arrival Route (STAR) in the TMA, and further execute the approach procedures. In order to achieve coordinated traffic control, we propose an integrated problem that tries to minimize the safety issues in both enroute phase and TMA simultaneously while considering the emergency level of the safety issues. Moreover, in real operations, the position and

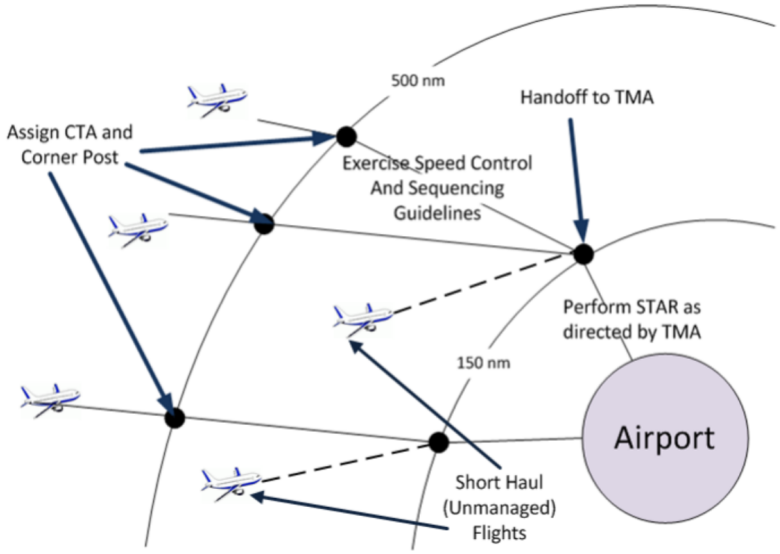


Figure 4.1: Procedures of the application of E-AMAN (Peters, 2017).

operating parameters of an aircraft keep changing with time, which increases the difficulty in solving the problem with a large range. In this context, the proposed algorithm makes periodic updates on the situation of the system including newly upcoming flights, flight locations, operational parameters of flights and the information related to uncertainty. In our problem, we assume aircraft are smoothly operated on the concerned horizon without executing holding and stacking in the terminal area. Therefore, aircraft trajectory prediction in the concerned range can be performed. Based on the trajectory information, controllers can have overall situation awareness with the help of decision support tool so as to solve potential safety issues.

4.2 Mathematical model

The model is established based on several assumptions and considerations. First, the network is divided into two main parts: the enroute phase and the TMA. According to the different control strategies, associated metrics are introduced for the two segments. Second, we address the important issues in a dynamic decision-making environment. Therefore, a dynamic framework is adopted and presented. Third, the model considers long term situations and discounts the safety issues that are far from the current time with a dynamic weighted approach.

4.2.1 Notations

The given data for a flight f in the arrival flight set $\mathcal{F}$ contains the following information:

- c_f: aircraft categories (heavy, medium, light).
- κ_f: flight type for identifying whether flight f is a long haul or short haul flight.
- n_f^b: boundary node, from which the flight f enters the E-AMAN horizon. For the short haul flights, this parameter represents the departure airport.
- $\mathcal{U}_f$: arrival route set of flight f in the cruising phase. Note that flight can be assigned to an alternate route from one of the waypoints in the enroute phase such that path stretching can be applied. Hence, multiple route options are included.
- h_f: the sequence order of a specific waypoint on the nominal route that aircraft f operated on. At this waypoint, an alternative route starts and deviates from the nominal route. For example, for f, if there are two routes can be chosen from its boundary node, then $h_f = 1$.
- t_f^b: boundary entry time of flight f.
- v_f^{opt}: nominal cruising speed obtained based on the general aircraft performance and environmental conditions, this speed is associated with the aircraft type.
- ν_f^e: TMA entry node of flight f.
- r_f: initial landing runway of flight f.

4.2.2 Network abstraction

It is known from the concept of E-AMAN, the flights that arrive at the same destination within a horizon of approximately 500NM are considered. Taking the Paris Charles de Gaulle (CDG) airport as an example, the trajectories of the arrival flights within 500NM from the airport are displayed in Fig. 4.2. The x-axis and y-axis display the longitude and latitude. In this figure, flights that come from the east side are gradually merged into the main stream through several waypoints, while flights that come from the west side follow a free routing strategy which can be directed from one waypoint directly to another. That is why the trajectories from the east side of the Paris CDG airport are more dispersed.

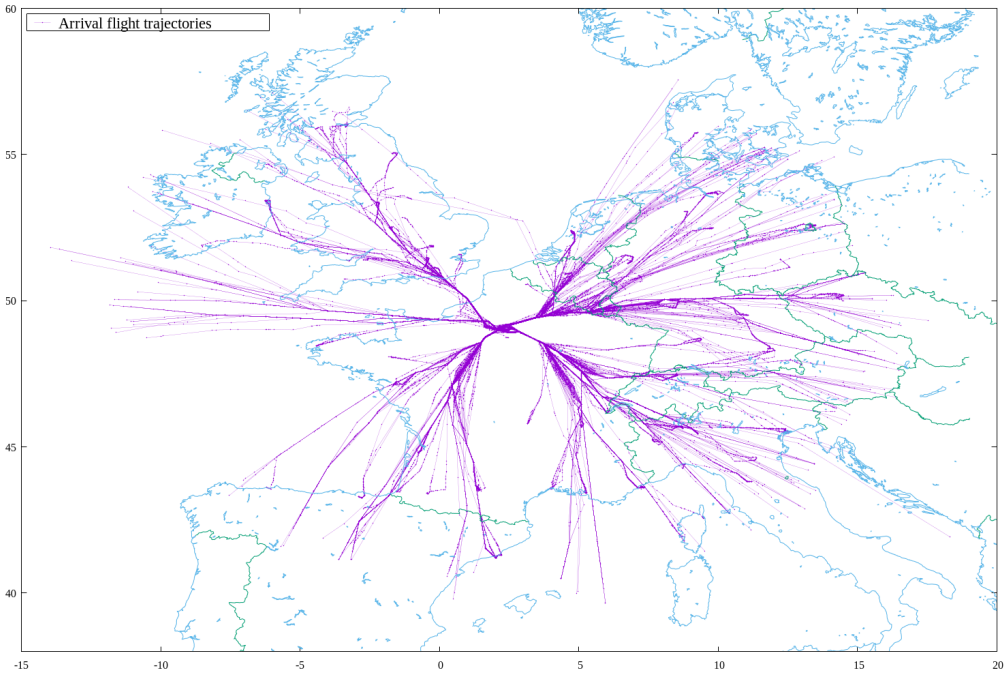


Figure 4.2: Trajectories of all flights arrive at the Paris CDG airport on Feb 10th, 2017. Only the trajectories within 500NM from the airport are displayed.

In order to simplify the arrival routes, routes that represent the main stream of traffic are abstracted as a network represented as a graph $\mathcal{G} = (\mathcal{N}, \mathcal{L})$ where $\mathcal{N}$ is the node set and $\mathcal{L}$ is the link set. In the network, $\mathcal{N}$ is constituted of several subsets, and we have $\mathcal{N} = \mathcal{N}_b \cup \mathcal{N}_c \cup \mathcal{N}_e \cup \mathcal{N}_t \cup \mathcal{N}_r$, where $\mathcal{N}_b$ denotes the node set of boundary waypoints, $\mathcal{N}_c$ represents the node set in the enroute segment, $\mathcal{N}_e$ stands for the node set of metering fixes which are the TMA entry nodes, $\mathcal{N}_t$ represents the nodes extracted from the arrival route in the approach phase, and $\mathcal{N}_r$ denotes the set of runway thresholds. We assume that the network is divided into two main regions, the enroute network and the terminal network, taking $\mathcal{N}_e$ as the reference points to enter the TMA from enroute phase. The nodes are connected by links. In line with the predefined network regions, the link set is defined as $\mathcal{L} = \mathcal{L}_c \cup \mathcal{L}_t$, where $\mathcal{L}_c$ and $\mathcal{L}_t$ denote the link set of the enroute segment and the link set of the TMA, respectively.

Each route $u \in \mathcal{U}$ is composed of two sub-routes u^e and u^t which refers to the route in the enroute segment and in the TMA respectively. In the enroute segment, u^e is composed

of a set of nodes $(n_1^u, n_2^u, ..., \nu_e)$, and a set of links $(l_1^u, l_2^u, ...)$ that connect those nodes from the boundary until the TMA entry node $\nu_e \in \mathcal{N}_e$. In the node set, $n_b \in \mathcal{N}_b$ is the first node that an aircraft passed to enter the system, therefore we can also replace n_1^u to n_b^u. We denote n_i^u as the i-th node that an aircraft would pass by following the route u. For links, we have $l_i^u \in \mathcal{L}_c$ with i indicating the sequence order of this link on route u. Similarly, a route u^t in the TMA consists of a set of nodes $(\nu_e, \nu_1, \nu_2, ..., \nu_r)$ and a set of links $(\mu_1, \mu_2, ...)$, where $\nu_r \in \mathcal{N}_r$ and $\mu_i \in \mathcal{L}_t$. The TMA route is uniquely defined by specifying the TMA entry node and a landing runway. Each aircraft follows exactly one route to arrive at the airport. An example of the notations representing different resources on a random route is illustrated in Fig. 4.3. In addition, the abstracted network of the whole system is shown in Fig. 4.4, including the networks of the enroute segment and the TMA. In order to better simulate the aircraft operation process, the x-axis and y-axis represent the distance with a unit of nautical mile with respect to a specific reference point. In this study, we assume that the speeds of aircraft keep constant on each link in the enroute phase, and once aircraft enter the TMA, a constant deceleration operation is performed.

4.2.3 Decision variables and constraints

Regarding the geographical structure and operational constraints of the cruise phase (enroute) and the approach phase (TMA), decision variables are specified accordingly. From a practical operational point of view, decision support tools provide advisories for air traffic controllers taking into account the information related to aircraft or environment. Therefore, we take the trajectory adjustment instructions as decision variables.

Decision variables for enroute segment

Considering the instructions that the air traffic controllers usually give to the pilots, two kinds of decision variables are applied in the enroute segment, which are:

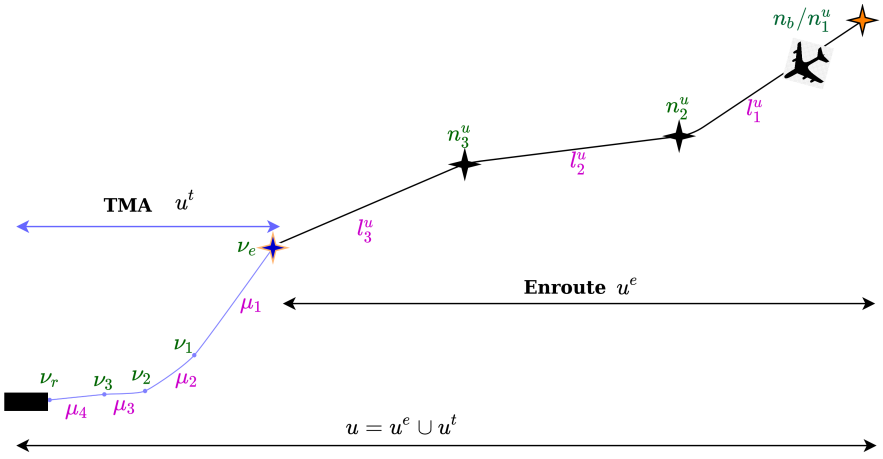


Figure 4.3: Example of the resources notations of a random route from a boundary entry point until a runway threshold.

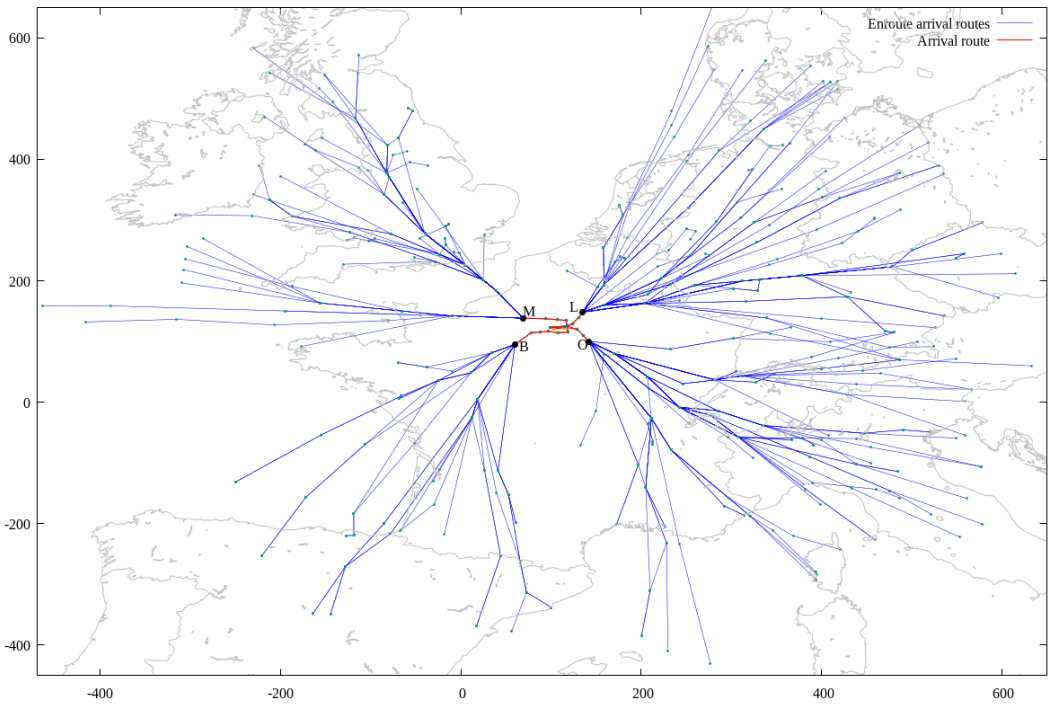


Figure 4.4: Arrival routes network abstraction for the enroute phase and the TMA within 500NM from the destined airport. Enroute network and TMA network are indicated in blue and red, respectively.

- Alternative route u_f:

$$u_f \in \mathcal{U}_f \tag{4.1}$$

  An alternative route usually diverts from a waypoint which can increase or decrease aircraft flying distance in the cruising phase, leading to a gain or loss of time when arriving at the TMA. If an aircraft has not passed the waypoint that leads to an alternative route, the alternative route can be selected.

- Enroute link speed $v_{l_i}^{u_f}$, i is a sequential indicator representing the i-th link on route u_f that aircraft f passes through:

$$v_{l_i}^{u_f} \in V_f := \{v_f^{opt} + j\Delta_f^v | j \in \mathbb{Z} \text{ and } -10 \leqslant j \leqslant 10\} \tag{4.2}$$

  where Δ_f^v is a speed adjustment step that is set to be $0.01v_f^{opt}$. The enroute speed range is determined by taking into account the aircraft structural limit and the common speed range that are used for aircraft and in this study, it is chosen by referring to some literature. In Delgado and Prats (2012), a speed reduction concept that aims to absorb the delay in the cruising phase without increasing fuel consumption is proposed. Through a comprehensive study of parameters such as flight level, cost index, aircraft category, they stated that a speed reduction up to -12% can be achieved with respect to the nominal speed for certain types of flights. Moreover, Nikoleris *et al.* (2012) has mentioned that a speed reduction up to 10% of nominal cruising speed is the common expected range of speed reduction in the cruising

phase. It applied this speed reduction range in the cruising phase to investigate the ability of delay absorption of aircraft. Moreover, in Huang and Tomlin (2009), a speed variation between 12% to −10% of the nominal speed is applied to solve the conflicts in the sectors. Considering all these possible choices, in our problem, the speed constraints are set with a range indicated in Eq. 4.2.

Decision variables related to the TMA

Decision variables for the TMA can refer to our previous study that address the scheduling problem in the TMA (Huo *et al.*, 2021). One key factor that distinguishes decision variables selection from the previous study is that the margin of TMA entry time is not included. We assume that through enroute adjustment, required time transfer can be achieved in this model. That is to say, a smooth operation from the enroute phase to the arrival phase, and then to the approach is permitted without assigning holding and stacking time. Therefore, two decision variables are implemented for the TMA:

- TMA entry speed v_f^t:

$$v_f^t \in V_f^t := \{v_f^{optR} + j\Delta_f^{vR} | j \in \mathbb{Z} \text{ and } -10 \leqslant j \leqslant 10\} \tag{4.3}$$

  where v_f^{optR} denotes a nominal speed for flights when arriving at the TMA entry point and $\Delta_f^{vR} = 0.01\Delta_f^{optR}$.

- Landing runway r_f:

$$r_f \in \mathcal{N}_r \tag{4.4}$$

  During the approach phase, aircraft are assigned with different runways to ensure a sufficient usage of runway capacity.

The decision vector for all flights is represented as $\mathbf{x}$. The decisions vector for a flight f is denoted as $\mathbf{x}_f$, which contains all the decision variables of this flight: $\mathbf{x}_f = (u_f, \mathbf{v}_{l_i}^{u_f}, v_f^t, r_f)$, where $\mathbf{v}_{l_i}^{u_f}$ is a set of speeds of f associated with the enroute links. We denote the i-th link of aircraft f in the enroute phase when following the route u_f as $l_i^{u_f}$, which implies that route decision u_f of f should be specified first.

4.2.4 Metrics evaluation

The evaluation metrics proposed for the enroute phase and the TMA associates to the excess of airspace capacity and the separation losses. In this study, considering the operational characteristics in different phases, enroute metrics are set at a macroscopic level, while the metrics for the TMA are more tactical.

Traffic flow-based evaluation in the enroute segment

For our problem, only the arrival flights to a specific destination are concerned which is a part of the total enroute traffic. In this case, the published maximum capacity of a air traffic control sector is not suitable to use for congestion evaluation. Therefore, we propose a new traffic flow-based approach to evaluate the capacity excess of the arrival traffic in

the route network by counting the number of flights passing a specific link within a fixed time interval. In the process of traffic flow-based evaluation, Δ (e.g. 1 minute) is set as a time unit that constitutes the evaluation time interval which is set as $L = q \cdot \Delta$, where q is a coefficient to define the length of the evaluation time interval. For each time interval, the concerned events such as aircraft arriving at a specific link through its entry point are focused. As shown in Fig. 4.5, the horizontal lines starting from a time t_i indicates an evaluation time interval. The number of the concerned events in a time period $[t_i, t_i + L]$ are counted and denoted as $E(t_i)$. The events are represented by the black arrows located on the time horizon with respect to their occurring time. A continuous moving of the evaluation time interval is proceeded by shifting one time step Δ afterward.

Congestion evaluation

According to the above-mentioned flow-based evaluation method, the congestion of the links associated with the enroute segment is evaluated. The number of aircraft arriving at the link entry point in a time period L is counted, which can roughly represent the air traffic density in a link. For this approach, one important thing is to decide the length of the evaluation time interval. A proper time length not only can capture the traffic density variation but also avoid information loss caused by long time averaging. Empirically, q is set to 10, and the number of aircraft that enter link l from t to $t + L$ can be expressed as follows:

$$E_l(t) = \sum_{s=0}^{s=(q-1)} \sum_{f \in \mathcal{F}} \omega_{f,l}^{t+s\Delta} \qquad l \in \mathcal{L}_c, \tag{4.5}$$

where $\omega_{f,l}^{t+s\Delta}$ is a binary parameter that equals to 1 if flight f is predicted to enter or to exit link l at the time slot of $t + s\Delta, s \in [0, q-1]$; otherwise equals to 0. Taking Fig. 4.5 as an example, the black arrows in this figure can be interpreted as the events that the aircraft pass the entry point of link l with respect to the times, then the number of aircraft that enter link l during a time period L starting from time t_i are counted, in this example, $E_l(t_i) = E(t_i)$.

When surveillance systems are used, according to the minimum separation requirements prescribed by ICAO (2016), the horizontal separation standard between aircraft flying at the same altitude is 5NM in the controlled enroute airspace. Based on which, the

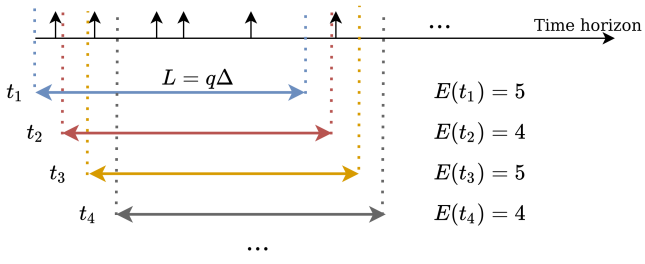


Figure 4.5: Traffic flow-based evaluation with the time period rolling scheme to evaluate enroute congestion. The concerned events are represented by the black arrows, the number of the events corresponding to each time window is counted.

link capacity for a time period of L can be determined by transferring the distance based requirement to the capacity based one. The capacity is then defined as:

$$R_l = \frac{M \cdot v_l}{K} \tag{4.6}$$

where M refers to the separation constraint of one flight per 5NM. K is a parameter which is determined by computing the ratio of the time period L to one hour. As $L = 10$ mins, in this case, $K = 6$. v_l represents the average speed of all aircraft operated on link l. Let us define a congestion indicator $\phi_l(t)$, which is the excess number of aircraft over the capacity R_l for a time period from t to $t + L$.

$$\phi_l(t) = \max[(E_l(t) - R_l), 0], \quad l \in \mathcal{L}_t \tag{4.7}$$

If $E_l(t) - R_l$ is bigger than 0, congestion appears in the time period $[t, t + L]$ on link l. Otherwise, there is no congestion. The evaluations are carried out for all enroute links at all evaluation time intervals on the time horizon.

Resolution workload evaluation

In the arrival route network, apart from capacity management, another task of the air traffic controllers is to coordinate the traffic merging at specific waypoints. It is known that the number of aircraft that one controller can coordinate at a time is limited. Then the conflict resolution workload proposed in Delahaye and Puechmorel (2013) is applied. For a node where two or more links converge, the conflict resolution workload is evaluated based on the air traffic coming from the preceding links. These preceding links are called parent links of that node. Suppose that two links l_i and l_j are parent links of node n which belong to the parent link set $\mathcal{P}_n$ of node n. The average aircraft speeds on the links are given by v_{l_i} and v_{l_j} respectively. $E_{l_i}^{Out}(t)$ and $E_{l_j}^{Out}(t)$ are the numbers of flights that exit the parent links during the time interval $[t, t + L]$. The conflict resolution workload $E_n(t)$ for a time period $[t, t + L]$ can be computed as follows:

$$E_n(t) = \sum_{\substack{l_i, l_j \in \mathcal{P}_n \\ l_i \neq l_j}} \frac{2N_s\sqrt{v_{l_i}^2 - 2v_{l_i}v_{l_j}\cos\theta_{i,j} + v_{l_j}^2}}{v_{l_i}v_{l_j}\sin\theta_{i,j}K} \cdot E_{l_i}^{c,Out}(t)E_{l_j}^{c,Out}(t) \quad n \in \mathcal{N}_c, \tag{4.8}$$

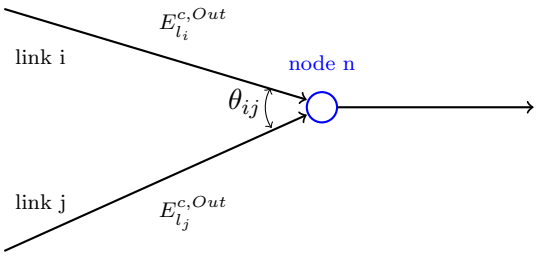


Figure 4.6: Conflict resolution workload evaluation configuration. The merging conflict is estimated based on traffic flows coming from the parent links and the angle that constructed by these two parent links.

where N_s is the standard separation norm which equals to 5NM, $\theta_{i,j}$ denotes the angle formed by two convergent flows associated with link l_i and l_j. In this equation, we first transfer flight mach number to a unit of NM/H. Moreover, K is used to convert the unit from hour to L (10 mins). Fig. 4.6 displays the general configuration when evaluating the conflict resolution workload, for which at least two parent links are required. If more than two parent links exist for a node, the conflict resolution workload is just the sum of the workloads induced by the possible pairs of the parent links. The indicator of the excess of the conflict resolution workload is given as:

$$\phi_n(t) = \max[(E_n(t) - R_n), 0], \quad n \in \mathcal{N}_c, \tag{4.9}$$

where R_n is the maximum conflict resolution workload. It is known that the workload is hard to be evaluated. In order to estimate a suitable value for R_n, we set up an upper limit by taking into account the aircraft speeds, maximum number of aircraft can exit from a link during L and a set of angles that appears in the network for evaluation, empirically, R_n is set to 1.34, namely, the potential conflicts of air traffic flows flying to the same point need to be lower than 8 for each hour.

Conflict detection in the TMA

The conflict detection approach in the TMA is identical to the model introduced in Chapter 3, where conflict detection on links, nodes, and runways correspond to three practical safety issues: the wake turbulence separation requirements, radar horizontal separation requirement and the runway operational separation requirements. In this approach, each pair of aircraft that consecutively operated on a resource are given a tailored minimum separation. This requirement is calculated according to aircraft operational parameters, the nominal separation requirements of the conflict type, and the geographical situation. We suppose, for two consecutively operated aircraft f and g, the conflict indicators of link, node and runway, are given identically as in Chapter 3.

- Node conflict indicator,

$$\phi^{\text{nc}}_{f,g,\nu} = \begin{cases} 1, & \text{if } 0 \le t_{g,v} - t_{f,\nu} \le \tau^{nc}_{f,g,\nu} \text{ or } 0 \le t_{f,\nu} - t_{g,\nu} \le \tau^{nc}_{g,f,\nu}, \quad \forall \nu \in u^t_f \cap u^t_g \cap \mathcal{N}_t \\ 0, & \text{otherwise} \end{cases} \tag{4.10}$$

- Link conflict indicator,

$$\phi^{\text{lc}}_{f,g,\mu} = \begin{cases} 1, & \text{if } 0 \le t_{g,\nu} - t_{f,\nu} \le \tau^{lc}_{f,g,\mu} \text{ or } 0 \le t_{f,\nu} - t_{g,\nu} \le \tau^{lc}_{g,f,\mu}, \quad \forall \mu \in u^t_f \cap u^t_g \cap \mathcal{L}_t, \\ 0, & \text{otherwise ,} \end{cases} \tag{4.11}$$

- Runway conflict indicator,

$$\phi^{\text{rc}}_{f,g} = \begin{cases} 1, & \text{if } 0 \le t_{g,r} - t_{f,r} \le \tau^{rc}_{f,g} \text{ or } 0 \le t_{f,r} - t_{g,r} \le \tau^{rc}_{g,f}, \qquad \text{if } r_f = r_g = r \in \mathcal{N}_r, \\ 0, & \text{otherwise} \end{cases} \tag{4.12}$$

The computation of the tailored separation requirements of a random flight pair (f, g) for link $\tau^{lc}_{f,g,\mu}$, node $\tau^{nc}_{f,g,\nu}$,and runway $\tau^{rc}_{f,g}$ are given in Appendix A.

4.2.5 Wind effects

Flights are always subject to uncertainties coming from themselves and the environment. In this study, we have considered the influence of wind, which is the most critical factor that affects the aircraft trajectory in the enroute phase by affecting the speed of aircraft. As a matter of fact, wind is a very dynamic factor. Even if there are accurate predictions, the frequent changes make the air traffic management complicated. In order to simulate the wind, it is characterized by randomly changing in a range of $v_W \in [v_W^{\min}, v_W^{\max}]$. The ground speed of the flights on each link is therefore a vector composition of aircraft true air speed vector and wind vector on the route direction:

$$\overrightarrow{v^G_{f,l}} = \overrightarrow{v_{f,l}} + \overrightarrow{v_W}. \tag{4.13}$$

where $\overrightarrow{v^G_{f,l}}$ is the ground speed vector on the route direction, $\overrightarrow{v_{f,l}}$ denotes the aircraft true air speed on the route direction, and $\overrightarrow{v_W}$ represents the wind component on the route direction. The ground speed is considered as the actual operational speed of each aircraft for the cruising phase and is used for trajectory prediction. Fig. 4.7 shows the relations of wind speed, true aircraft speed and ground speed.

4.2.6 Objective function

The objective function is set as a weighted sum of the previously introduced metrics including congestion and conflict resolution work in enroute segment and conflicts in the TMA. It is given as:

$$\begin{aligned} G(\mathbf{x}) = \alpha\Big\{ \sum_{t}^{T} \Big(\sum_{l \in \mathcal{L}_t} \phi_l(t) + \sum_{n \in \mathcal{N}_c} \phi_n(t) \Big) \Big\} \\ + \beta\Big\{ \sum_{\substack{f,g \in \mathcal{F}, \\ g>f}} \Big(\sum_{\nu \in u_f \cap u_g \cap \mathcal{N}_t} \phi^{\mathrm{nc}}_{f,g,\nu} + \sum_{\mu \in u_f \cap u_g \cap \mathcal{L}_t} \phi^{\mathrm{lc}}_{f,g,\mu} \Big) + \sum_{\substack{f,g \in \mathcal{F}, \\ g>f | r_f = r_g}} \phi^{\mathrm{rc}}_{f,g} \Big\} \end{aligned} \tag{4.14}$$

where α and β are the weights for the metrics associated with the enroute phase and the TMA, respectively.

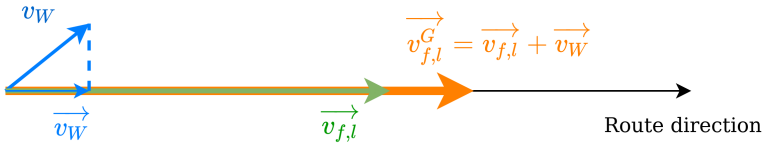


Figure 4.7: The relations of the ground speed, true airspeed and the wind speed, in which the wind speed direction is uncertain. The aircraft ground speed can be obtained by adding the wind speed vector projected in the same direction as the route.

4.2.7 Dynamic arrival management

It is known that flight operation is an ongoing and dynamic process subject to uncertainty. Air traffic controllers usually give instructions to the pilots according to the revised information both from the flights and the environment. Based on this fact, the newly available information needs to be involved in the assessment of future situations in order to achieve collaborative control. To realize this concern, we propose a dynamic control model that combines static sub-problem solving and dynamic weighted approach. Through these designs, reliable decisions can be provided to deal with recent problems while considering future situations.

Dynamic control framework

In a dynamic environment, the positions, speeds and directions of the aircraft that need to be managed are constantly changing. In order to have a high level of accuracy in the traffic prediction, it is necessary to all kinds information related to traffic management. A typical approach for such dynamic problem is the rolling horizon (sometimes is referred to as receding horizon) approach. The horizon is defined as a short time period to ensure that air traffic controllers have accurate knowledge of the overall situation in the system. Considering that the Air Navigation Service Provider (ANSP) would update the list of flights that have to be scheduled every 15-30 minutes (Jones *et al.*, 2013). Thus, we choose a time period of $\tau = 20$ mins as the length of the rolling horizon. After each τ units of time, the considered horizon is rolled, and the related information is updated. The time, at which the information update should be performed is referred to as *update time*. Suppose that t_s and t_e are the start time and the end time of the overall time horizon, then the k-th update time (k-th information update) corresponds to a current time of $t_c = t_s + (k-1)\tau$. Based on this framework, the algorithm is established by solving the sub-problems associated with each information update. Two key steps are involved to establish each sub-problem:

- Information update: This process includes specifying the list of flights that would be scheduled into the E-AMAN horizon within τ units of time. For aircraft that are already in the system, their locations and operational parameters are updated. Any aircraft that already landed at current time is removed from the system. Note that, together with other information, wind change is also taken into account.

- Aircraft trajectory prediction: This is carried out to achieve a comprehensive awareness of future air traffic situation based on the updated information. All aircraft in the system are simulated from their current locations until they land on the runway. The objective therefore can be obtained based on the predicted information. Sub-problem are then established aiming at generating suitable control decisions to ensure safe and efficient air traffic operation for next τ units of time. We denote a prediction horizon t_r as the longest time required for all aircraft in the system to reach the runway threshold from their current positions. This value also comes from prediction.

Fig. 4.8 illustrates the rolling horizon approach. The vertical dash lines on the time

horizon indicate the update time. For each update time, the prediction horizon t_r is varied and computed based on the aircraft operational parameters including the decisions obtained from the optimization of the previous sub-problem and the current aircraft locations.

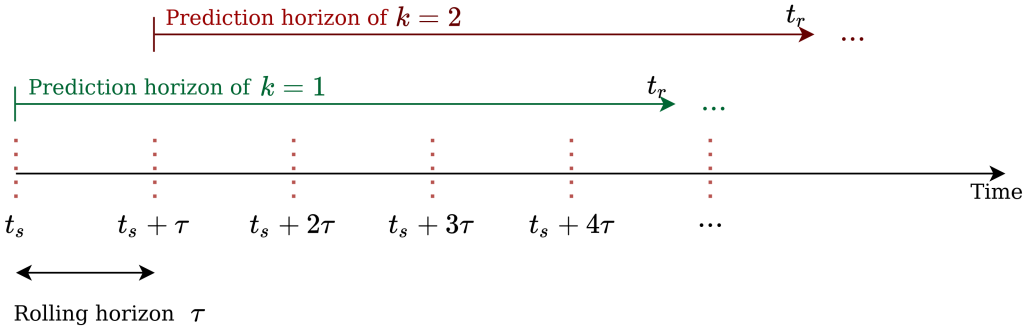


Figure 4.8: Dynamic framework with rolling horizon approach. Rolling horizon has a length of τ. Each horizon rolling functions as a time indicator, for which information update and aircraft trajectory prediction are carried out. Prediction horizon is an unfixed value computed based on the operational parameters and current locations of the aircraft in the system.

Another critical concern is the determination of the freezing zone. In some literature, the freezing zone is set with time (Murça and Müller, 2015; Bennell *et al.*, 2017). Considering the context of our problem, the freezing zone is chosen as a geographical area, in which the decisions of aircraft are fixed. In our model, the speed profile of each aircraft in the TMA only depends on its two decision variables: TMA entry speed and the runway. These two decision variables are determined before an aircraft gets into the TMA, therefore the terminal area can be naturally considered as the freezing zone. Note that at any update time, some aircraft are operated in the TMA. These aircraft are equally simulated and involved in the conflict detection process in order to guarantee overall safety.

In the rolling horizon framework, it is necessary to define a parameter for each aircraft that can explicitly indicate its flying progress in the network. It is worth mentioning that in the system, we assume that the control speeds of an aircraft are assigned for each link. Therefore, it is more practical to associate the aircraft position with a link instead of obtaining an exact distance or coordinate. Considering that we are now at the k-th update time with a current time $t_s + (k-1)\tau$, we denote p_f^k the progress parameter of an aircraft f, which can be interpreted as the sequence order of the link that f is currently on, with respect to the chosen route. In summary, aircraft can be classified according to the following states:

- Pre-processing: Aircraft that will pass the boundary entry node within τ units of time from the current update time. All aircraft in the pre-processing state constitute the flight set $\mathcal{F}_p$. Since aircraft in this state have not operated on any link in the concerned horizon, for an arbitrary aircraft f belonging to $\mathcal{F}_p$, its progress parameter at the k-th update time is:
$$p_f^k = 0, \quad f \in \mathcal{F}_p, \tag{4.15}$$
- Enroute: Aircraft that have passed the boundary entry node but did not entered the

TMA yet belong to the enroute flight set $\mathcal{F}_e$. In this state, aircraft trajectories are predicted from their current positions to their landing runways based on the control inputs obtained from previous optimizations. We then consider a model that can explicitly express the progress transition between time $t_s+(k-2)\tau$ ($k-1$-th update time) to $t_s+(k-1)\tau$ (k-th update time) of aircraft f:

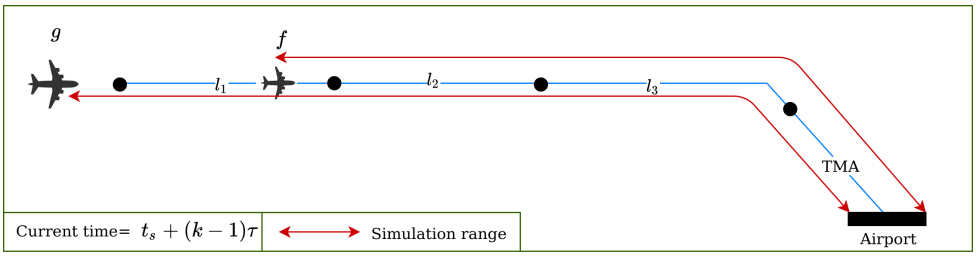
$$p_f^k = y(p_f^{k-1}, \mathbf{x}_f^{k-1}), \quad f \in \mathcal{F}_e, \tag{4.16}$$

where $y(.)$ represents the real operation process during the period of τ units of time, and $\mathbf{x}_f^{k-1}$ is the decision variables obtained from the previous optimization ($k-1$-th sub-problem). Since the lengths of the links are different, it is possible that p_f^{k-1} equals to p_f^k.

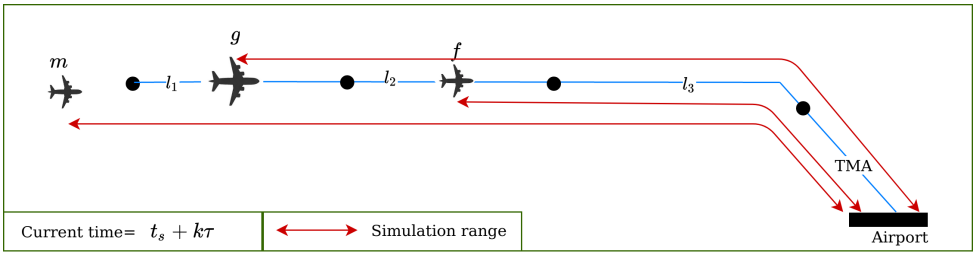
- Approach: aircraft that have passed the TMA entry node and operated in the TMA are put in the approach flight set $\mathcal{F}_t$. We consider the operation of an aircraft in the TMA as the last state. Since the decision variables related to the operation in the TMA are determined before the aircraft get in the TMA. From a macroscopic point of view, the route from the TMA entry node to the runway can be seen as one last link. Once an aircraft gets out of the runway, its progress parameter is set to -1, which means it is out of the system and is no longer concerned.

In order to clearly explain the dynamic control frame, a simplified example is given with one arrival route and several consecutively operated aircraft. Fig. 4.9 shows the situations of two consecutive rolling horizons. We suppose that aircraft f, g and m enter the E-AMAN horizon from the same boundary entry node. As shown in Fig. 4.9a, at k-th update time, aircraft f is already in the system and aircraft g will enter the boundary within τ units of time, therefore, $f \in \mathcal{F}_e, g \in \mathcal{F}_p$. In this case, $p_f^k = 1$ and $p_g^k = 0$, which means f is on the first link of its route and g has not entered the system yet. Based on the current positions of aircraft and the decision inputs, the predicted trajectories of all aircraft in the system that start from their current positions until landing are obtained. The sub-problem associated with the situation of current update time is formulated. By minimizing the objective, control decisions are determined for all aircraft in the system. Specifically, for f and g, after the optimization of the k-th sub-problem, speeds on their next links (with associated sequence order of p_f^{k+1} and p_g^{k+1}) are obtained and will be delivered to pilots before f arrives at link $l_{p_f^{k+1}}$ and g arrives at $l_{p_g^{k+1}}$. The speeds are denoted as $v_{l_{p_f^{k+1}}}^{u_f}$ and $v_{l_{p_g^{k+1}}}^{u_g}$.

After τ units of time passed, the locations of aircraft f and g are updated, as shown in Fig. 4.9b, from which we know that $p_f^{k+1} = 2$ and $p_g^{k+1} = 1$. The newly scheduled aircraft $m \in \mathcal{F}_p$ will enter the boundary at required entry time, based on which its trajectory prediction is made. $k+1$-th sub-problem is then established based on the predicted trajectory information of all aircraft in the system. The optimal decisions that minimize safety issues will be assigned to all aircraft. For the three aircraft the decisions need to be delivered are $\mathbf{x}_m^{k+1} = (u_m, v_{l_{p_m^1}}^{u_m}, v_m^t, r_m)$ for flight m, $\mathbf{x}_g^{k+1} = (u_g, v_{l_{p_g^{k+2}}}^{u_g}, v_g^t, r_g)$ for flight g, and $\mathbf{x}_f^{k+1} = (u_f, v_{l_{p_f^{k+2}}}^{u_f}, v_f^t, r_f)$ for flight f.



(a) The simulation ranges for aircraft in the system at time $t_s + (k-1)\tau$, which is the k-th update time.



(b) The simulation ranges for aircraft in the system at time $t_s + k\tau$, which is the $(k+1)$-th update time.

Figure 4.9: Dynamic control process. At each update time, aircraft trajectories are predicted from their starting location until landing and the corresponding sub-problem is established.

Dynamic weighted approach

During the flight operation, aircraft in the system could be located anywhere in the route network. The aircraft that are far from the TMA have potential ability to impact the further situations by changing their decisions, whereas aircraft that are sufficiently close to the TMA can only affect the future situation inside the TMA. For aircraft that are far away from the TMA, the decisions assigned to them should always give priority to solving the short-term problems rather than those in the future. In addition, as an aircraft is getting close to TMA, its situation in TMA gets more and more important. Therefore, unified weights for the metrics of the enroute segment and the TMA are not realistic and not effective for a dynamic problem. We then propose a dynamic weighted approach that weights the metrics associated with enroute segment and the metrics associated with the TMA for each aircraft in accordance with its location. For simplicity, we refer to these two weights as enroute weight and TMA weight. For an aircraft, the enroute weight and TMA weight are determined based on the distance that needs to be proceeded from the next link. The sum of the enroute weight and TMA weight of an aircraft is 1. The weight assigned to the metrics related to enroute phase at the k-th update time for a random aircraft f is denoted as α_f^k, and it is computed as follows:

$$\alpha_f^k = \begin{cases} \frac{d_f^{e,k}}{D_f^k}, & \text{if } d_f^{e,k} > 0, \\ 0, & \text{if } d_f^{e,k} = 0, \end{cases} \tag{4.17}$$

where D_f^k and $d_f^{e,k}$ are the total distance and the enroute distance that need to be

proceeded by aircraft f. In order to be consistent with the decision making process, for an aircraft that is currently at the enroute phase, the valid distance starts from the next waypoint that the aircraft will pass until the runway threshold. Fig. 4.10 illustrates the distance for computing the weights allocation for an aircraft f, in which d_f^{tma} is the route length in the TMA based on its originally assigned runway. In this figure, aircraft f is at the second link on the route, $d_f^{e,k}$ is then the summed distance of the third link and fourth link of its current route. The total distance D_f^k is the sum of the enroute distance $d_f^{e,k}$ and the distance in the TMA, which is d_f^{tma}.

Considering that metric evaluations usually involve multiple aircraft, the weights of a single aircraft should be transferred to the weight of a metric so as to be incorporated into the objective function.

- **Weight of congestion evaluation in enroute segment**
  We denote $A_l^k(t)$ the weight for the congestion evaluation of link l at time interval $[t, t+L]$ for k-th sub-problem. This weight is simply an average of enroute weights of aircraft involved in a specific congestion evaluation. The value of the weight is given as:

$$A_l^k(t) = \frac{\sum_{s=0}^{s=q-1} \sum_{\substack{f \in \mathcal{F}_p \cup \mathcal{F}_e \\ l \in u_f}} \omega_{f,l}^{t+s\Delta} \alpha_f^k}{E_l(t)} \qquad l \in \mathcal{L}_c, \tag{4.18}$$

  We remind that $\omega_{f,l}^{t+s\Delta}$ equals to 1, if aircraft f gets in link l at time slot $t+s\Delta$, and $E_l(t) = \sum_{s=0}^{s=(q-1)} \sum_{f \in \mathcal{F}_p \cup \mathcal{F}_e} \omega_{f,l}^{t+s\Delta}$ is the number of aircraft that pass the entry point of link l during time period $[t, t+L]$. α_f^k is the enroute weight of aircraft f.

- **Weight of conflict resolution workload in enroute segment**
  The weight of the conflict resolution workload is computed by averaging the enroute weights of aircraft involved in the associated evaluation. Considering an evaluation on node n at a specific time interval $[t, t+L]$, the aircraft getting out of its parent links are involved. The weight is given as:

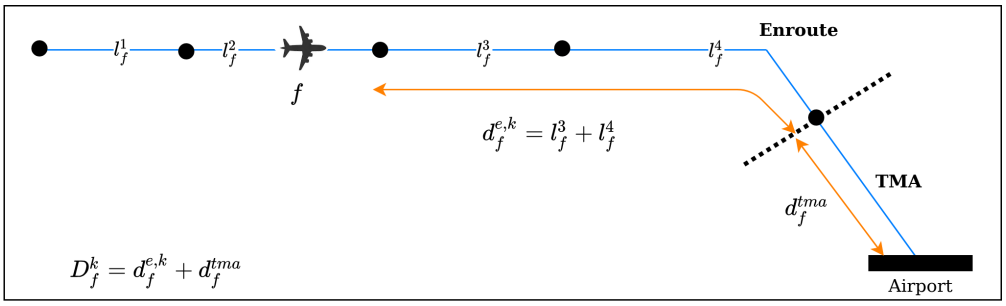


Figure 4.10: The valid distances for dynamic weighted approach. For the enroute phase, the distance from the next waypoint until TMA entry point is considered. The length used for the TMA weight starts from the TMA entry point until the runway threshold.

$$A_n^k(t) = \sum_{\substack{l_i, l_j \in \mathcal{P}_n \\ l_i \neq l_j}} \frac{\sum_{s=0}^{s=q-1} (\sum_{\substack{f \in \mathcal{F}_p \cup \mathcal{F}_e \\ l_i \in u_f}} \omega_{f,l_i}^{t+s\Delta} \alpha_f^k + \sum_{\substack{g \in \mathcal{F}_p \cup \mathcal{F}_e \\ l_j \in u_g}} \omega_{g,l_j}^{t+s\Delta} \alpha_g^k)}{E_{l_i}^{Out}(t) + E_{l_j}^{Out}(t)} \quad n \in \mathcal{N}_c, \tag{4.19}$$

Remind that $\omega_{f,l_i}^{t+s\Delta}$ (resp. $\omega_{g,l_j}^{t+s\Delta}$) equals to 1, if aircraft f gets out of the link l_i at time slot $t + s\Delta$. Correspondingly, we have $E_{l_i}^{Out}(t) = \sum_{s=1}^{s=q} \sum_{\substack{f \in \mathcal{F}_p \cup \mathcal{F}_e \\ l_i \in u_f}} \omega_{f,l_i}^{t+s\Delta} \alpha_f^k$ (resp. $E_{l_j}^{Out}(t)$) represents the number of aircraft that get out from link l_i (resp. link l_j) during time interval $[t, t + L]$.

- **Weight of conflict in the TMA**
  For conflict detection in the TMA, two aircraft are involved, and the weight of the conflict is the average weight of the TMA weights associated with these two aircraft. Suppose we evaluate the conflict between aircraft f and g, the associated weight for their conflict is:

$$\mathcal{A}_{f,g}^k = \frac{2 - \alpha_f^k - \alpha_g^k}{2}, \quad u_f^t \cap u_g^t \neq \varnothing \tag{4.20}$$

  We remind that α_f^k (resp. α_g^k) is the enroute weight of aircraft f (resp. g). Correspondingly, $1 - \alpha_f^k$ (resp. $1 - \alpha_g^k$) is the TMA weight of aircraft f (resp. g).

Fig. 4.11 provides a diagram of the dynamic optimization process. By applying the rolling horizon approach, the algorithm periodically executes several procedures including current time and information updating, aircraft trajectory simulations (prediction) to establish sub-problems. The control decisions can be obtained through optimizing the dynamically weighted objective functions. The pseudo-code of the dynamic optimization procedure is given in Algorithm 4.

4.2.8 Objective function of the dynamic model

The objective that we want to minimize for the k-th sub-problem is given as:

$$\begin{aligned} G(\mathbf{x}^k) = & \sum_{t=t_s+k\tau}^{t_r} \left(\sum_{l \in \mathcal{L}_c} A_l^k(t) \phi_l^k(t) + \sum_{n \in \mathcal{N}_c} A_n^k(t) \phi_n^k(t) \right) \\ & + \sum_{\substack{f,g \in \mathcal{F} \\ g > f}} A_{f,g}^k \left(\sum_{\nu \in u_f \cap u_g \cap \mathcal{N}_t} \phi_{f,g,\nu}^{\mathrm{nc}} + \sum_{\mu \in u_f \cap u_g \cap \mathcal{L}_t} \phi_{f,g,\mu}^{\mathrm{lc}} + \sum_{r_f = r_g} \phi_{f,g}^{\mathrm{rc}} \right) \end{aligned} \tag{4.21}$$

where the first term represents the weighted metrics associated with the enroute phase, and the second term is the weighted number of conflicts in the TMA. Solving the k-th sub-problem provides the optimal decisions for aircraft operation in next rolling horizon.

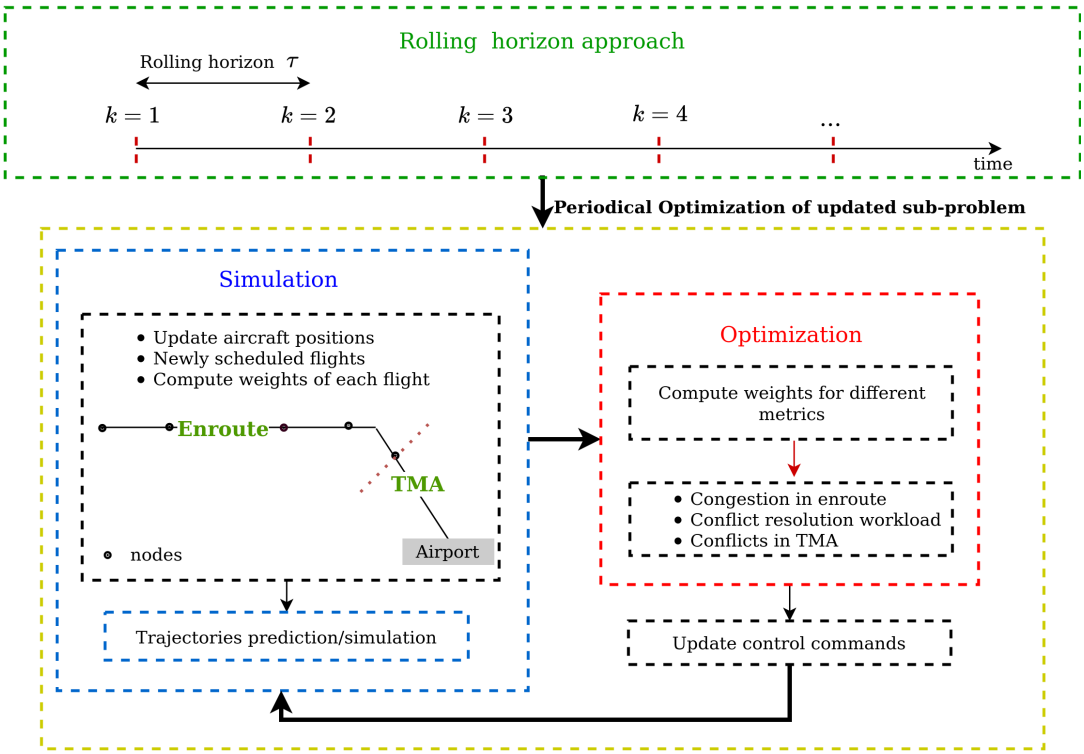


Figure 4.11: Dynamic optimization framework based on the rolling horizon approach. The rolling horizon approach functions as a time indicator to conduct information update. Based on the updated information, aircraft trajectory predictions are conducted and associated sub-problems are established. Output of the previous sub-problem is part of the input of current problem.

4.3 Solution algorithm

Considering the complexity of the E-AMAN system, Simulated Annealing (SA) is adapted to solve this problem under the proposed dynamic control environment. It is mentioned that the dynamic feature is captured through periodical information update, based on which sub-problems are established accordingly. SA is then applied to solve each sub-problem. Several critical procedures need to be specified:

- Neighborhood selection: This procedure plays a critical role in finding potential better solutions. In our problem, neighborhood selection refers to finding suitable control decisions for aircraft operations. It is known that the selection of alternative route is also an option to adjust enroute operation. One limitation is that only when an aircraft does not pass the node that separates the main route and the alternative route, the decision of alternative route can be changed. Consequently, in each sub-problem, the neighborhood selection process for an aircraft needs to first specify whether this aircraft can choose the alternative route at its current position or not. Fig. 4.12 provides an example to explain the condition of alternative route selection. In this figure, a route u and its alternative route u' are displayed. The node that separates the alternative route and the nominal route is marked in red. Aircraft g is now at a position before the red node and it can be directed to any of the available

Algorithm 4 Dynamic optimization problem incorporating rolling horizon approach, which is based on solving static problems every τ units of time.

Require: get t_s, t_e;
1: $k \longleftarrow 1$;
2: **while** $t_s + (k-1)\tau <= t_e$ **do**
3: $t_c \longleftarrow t_s + (k-1)\tau$;
4: Get pre-processing flight set $\mathcal{F}_p$ based on t_c and assign $p_f^k = 0$, $f \in \mathcal{F}_p$;
5: **for** $f \in \mathcal{F}_p \cup \mathcal{F}_e \cup \mathcal{F}_t$ **do**
6: Update aircraft progress parameter based on decision inputs, $p_f^k = y(p_f^{k-1}, \mathbf{x}_f^{k-1})$;
7: Compute the weight α_f^k;
8: Predict the trajectories of f based on the decisions $\mathbf{x}_f^{k-1}$ obtained from the previous sub-problem .
9: **end for**
10: **for** $l \in \mathcal{L}_c; n \in \mathcal{N}_c$ **do**
11: Compute the weights for the enroute metrics $A_l^k(t)$ and $A_n^k(t)$ at each $t \in (t_c + i\Delta | i \in [0, (t_r - t_c)/\Delta])$
12: **end for**
13: Compute the objective function $G(\mathbf{x}^k)$;
14: Execute the optimization procedure: $SubProblemOptimization(G(\mathbf{x}^k), \mathbf{x}^{k-1})$;
15: **for** $f \in \mathcal{F}_p \cup \mathcal{F}_e$ **do**
16: Obtain new decisions: $\mathbf{x}_f^k = (u_f, \mathbf{v}_f^{u_f}, v_f^t, r_f)$, $\mathbf{v}_f^{u_f} = (v_{l_1}^{u_f}, ..., v_{l_{p_f^{k+1}}}^{u_f})$;
17: **end for**
18: Obtain new decision inputs $\mathbf{x}^k$;
19: $k \longleftarrow k + 1$;
20: **end while**

routes. Instead, aircraft f has passed the red node, therefore, it has to follow the assigned route.

Apart from the decision of alternative route, other decision variables also need to be considered. A neighborhood solution is generated by randomly choosing one aircraft and changing one of its decision variables within the defined range. For each aircraft, the values of its enroute weights and TMA weights bias the choice of the decision variables. For example, if an aircraft has a high enroute weight, which means it is far away from the TMA, changing the decisions such as enroute speed and assigning alternative route are likely to be more reasonable. Whereas, for an aircraft near TMA, higher probability is allocated to change the decision variables such as TMA entry speed and runway. Algorithm 5 describes the neighborhood solution selection method for a sub-problem based on a random selecting strategy. We remind that h_f denotes the sequence order of a node, at which the alternative route is diverted from the main route.

- Stopping criterion: The algorithm terminates in two conditions. First, when the temperature reaches the lowest acceptable temperature. This temperature is set to be $10^{-4}T_o$, where T_o is the initial temperature for the cooling process. Second, the optimal solution is achieved (objective function equals 0).

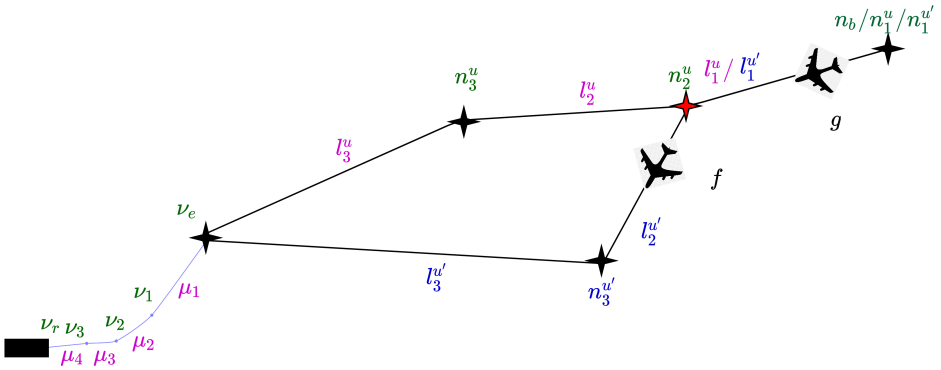


Figure 4.12: Example of alternative route selection condition. Aircraft located before the node that separates the alternative route and the nominal route can be assigned to either route. Aircraft that has passed that node needs to follow the assigned route, and the decision of route is fixed.

Algorithm 5 Neighborhood selection design for k-th sub-problem. For a chosen flight, the decision selected to be changed is biased by its weight. Aircraft positions need to be specified to determine whether the alternative route can be chosen.

Require: : Obtain the phase p_f^k of the each aircraft $f \in \mathcal{F}_p \cup \mathcal{F}_e$.

Generate random number, $a \in (0, 1)$;
Randomly choose one aircraft $f \in \mathcal{F}_p \cup \mathcal{F}_e$;
if $a <= \alpha_f^k$ **then**
    if $p_f^k < h_f$ **then**
        Randomly choose a decision variable from u_f and $v_{l_{p_f^{k+1}}}^{u_f}$, change its value within the possible range.
    else
        Randomly choose one value for $v_{l_{p_f^{k+1}}}^{u_f}$.
    end if
else
    Randomly choose either v_f^t or r_f, change its value within the possible range.
end if

4.4 Results

In this section, the case study and the analysis of the associated results are presented.

4.4.1 Case study

The algorithm has been tested on the flights arriving at the Paris CDG airport on 10th February 2017. Fig. 4.13 displays the number of arrival aircraft in the system with respect to time based on the real traffic data, from which we select the period from 5.AM to 1.PM to investigate the performance of our proposed model. The chosen period is indicated with a grey background. From Fig. 4.4, we know that arrival traffic comes from various directions and finally converges to four TMA entry points. The enroute network

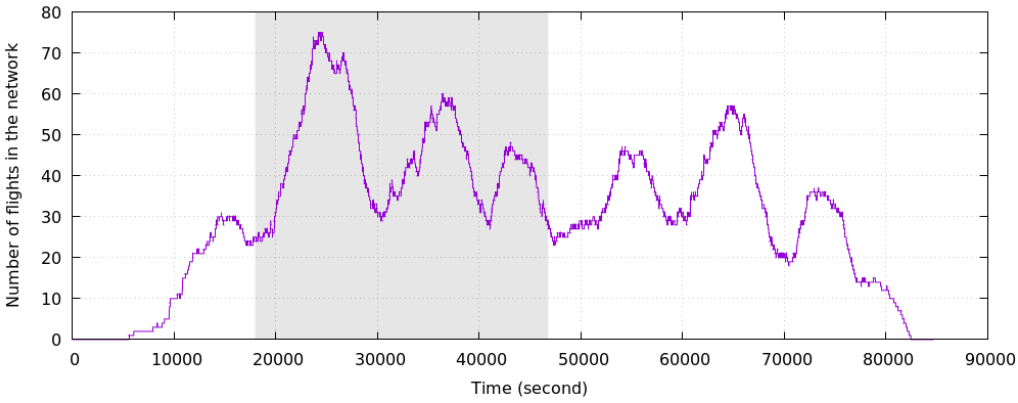


Figure 4.13: The number of flight with respect to time in the E-AMAN horizon destined to the Paris CDG airport on 10th February 2017. Grey shadow indicates the period with high traffic density.

information associated with the four points is given in Tab. 4.1 including the number of boundary entry nodes, number of routes, number of waypoints and number of links that are abstracted to establish the network.

Table 4.1: The information of the enroute arrival network of the Paris CDG airport within a range of 500NM.

	Nb of boundary entry nodes	Nb of routes	Nb of waypoints	Nb of links
MOPAR	38	43	78	83
LORNI	43	56	103	116
OKIPA	41	55	96	110
BANOX	14	25	42	52
Total	136	179	319	361

In addition, the information regarding the arrival flights in the chosen period associated with the four TMA entry points is given in Tab. 4.2. All flights involved in this period are heavy and medium with an overall mix ratio of heavy and medium that equals to 0.27.

Table 4.2: Arrival traffic information associated with the four TMA entry points.

	Medium	Heavy	Long haul	Short haul
MOPAR	41	40	42	39
LORNI	75	12	44	43
OKIPA	84	7	34	57
BANOX	43	7	13	37
Total	243	66	133	176

As for the routes in the TMA, the west-flow runway configuration (26L, 27R | 26R, 27L) is applied in this problem and the network abstracted for the TMA is identical to the previous chapter and it is presented in the network shown in Fig. 4.4 as well.

4.4.2 Results analysis

In our problem, the results are analyzed from two points of view. First, we investigate the metrics that constitute the objective during the optimization process for each sub-problem. Second, the numerical analysis of time transfers at TMA entry point is displayed. In our study, we use enroute trajectory adjustment to meet the required TTL and TTG for a smooth operation of aircraft in the TMA. The achieved time margins at the TMA entry point show the ability of enroute trajectory adjustment, based on which multiple benefits are realized.

Analysis of objectives

The metrics of enroute congestion, conflict resolution workload, and conflicts in the TMA represents the safety level of the system. One possible way to evaluate the performance of the algorithm is to investigate the variation of metric values during optimization. Fig. 4.14 shows the evolution of the non-weighted metrics including the total value of all the metrics and the three types of conflicts in the TMA. The congestion and the conflict resolution workload are not included in this figure, since compared to the value of conflicts, these two values are much smaller and hard to show in the same figure. This is because normally, a route can be provided to aircraft with different destinations, while in this study, we only focus on the air traffic destined to one airport. In this figure, the optimization processes of all sub-problems are shown, and we can observe that the metrics can be effectively minimized even with high initial values. However, for some sub-problems, the final objective values do not achieve 0, which means potential safety issues still exist. These sub-problems and the associated metrics values are given in Tab. 4.3.

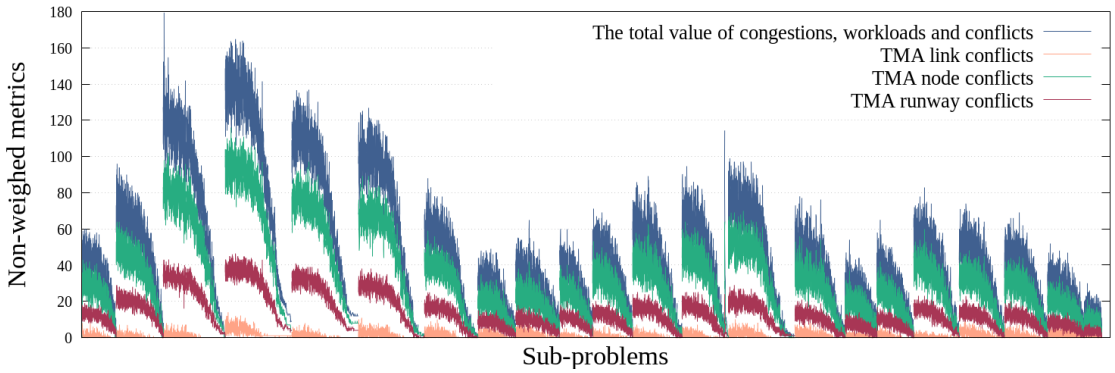


Figure 4.14: The evolution of the total value of all metrics and the number of link conflict, node conflict and runway conflict in the TMA of all sub-problems. The objective values can be significantly reduced in each sub-problem.

Table 4.3: The final results of the sub-problems whose objective values did not achieve 0. The final value of non-weighted metrics are given.

kth sub-problem	Congestion	workload	Link conflict	Node conflict	Runway conflict	Total
k=6	0	0	1	4	4	9
k=7	0	0	0	8	4	12
k=17	0	0	0	0	1	1

Table 4.4: For each sub-problem, the safety issues that occur within τ units of time from the associated update time are counted. The real safety issues that are not eliminated are given and the associated sub-problems are indicated.

kth information update	Congestion	workload	Link conflict	Node conflict	Runway conflict	Total
k=6	0	0	1	0	0	1
k=7	0	0	0	2	0	2
k=17	0	0	0	0	1	1

It is worth mentioning that, in each sub-problem, both the short-term and long-term situations are concerned by incorporating all the predicted information of aircraft in the objective evaluation. This consideration range can be up to several hours for some aircraft. However, in our study, regular optimizations based on updated flight information ensure that air traffic controllers issue the latest optimized decisions to pilots every τ units of time. This means that the control decisions are valid for around τ units time, and then new decisions will be issued. Therefore, the optimized values of the metrics within τ units of time from each update time indicate the real operational situation in the system are investigated. Fig. 4.15 illustrates the variation of the total value of all metrics and the values of three types of conflicts in the TMA that occur within τ units of time from each update time. The associated final values that do not achieve 0 are given in Tab. 4.4, in which at most 2 conflicts in the TMA are not solved in the sub-problems.

Note that Fig. 4.14 illustrates the metric variation associated with the prediction range t_r, while Fig. 4.15 shows only the safety issues arising within τ units of time from the information update and optimization. Differences can be observed in the figures. First, the range of the y-axis is higher in Fig. 4.14 than in Fig 4.15. Second, it is shown that the length on x-axis for each sub-problem is consistent. Compared the objective reduction process in the two figures, most of the metrics eliminated earlier in Fig. 4.15 than in Fig. 4.14, which implies that the short-time safety issues are addressed with priority and long-term safety issues are optimized later.

Observing the values in Tab. 4.3 and in Tab. 4.4, safety issues only happen in the TMA, and only the 6th, 7th and 17th sub-problem did not eliminate the conflicts. As we mentioned previously, the safety issues arising within τ units of time can be considered as the real safety issues need to be concerned. Thus, results shown in Tab. 4.4 means that there are one link conflicts, two node conflicts and one runway conflict occurring in the 6th, 7th and 17th sub-problem respectively. Specific measures can be applied by controllers at the tactical level to solve these conflicts, such as route stretching, holding, etc.

Regarding the computational efficiency, Fig. 4.16 gives the information in terms of the numbers of aircraft and the computation times of the associated sub-problems. Generally speaking, the number of aircraft involved in the problem is positively correlated with computation time. However, in the 15th sub-problem, the solution is found very fast. This is due to the randomness of the solution approach, since the algorithm randomly explores the state space.

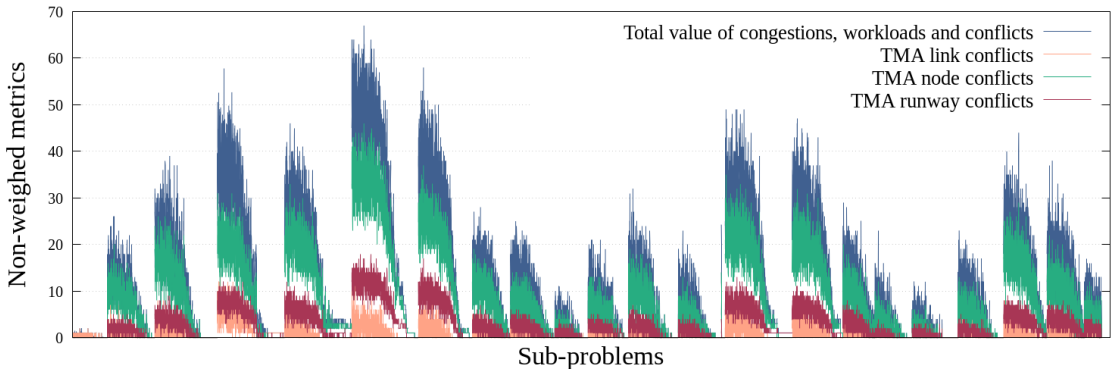


Figure 4.15: The evolution of the total value of all metrics and the number of link conflict, node conflict and runway conflict in the TMA arising within τ units of time from the update time of all sub-problems. The metrics are reduced nearly to 0 for each sub-problem. The final values of the metrics for the sub-problems show the real operational situation in the system.

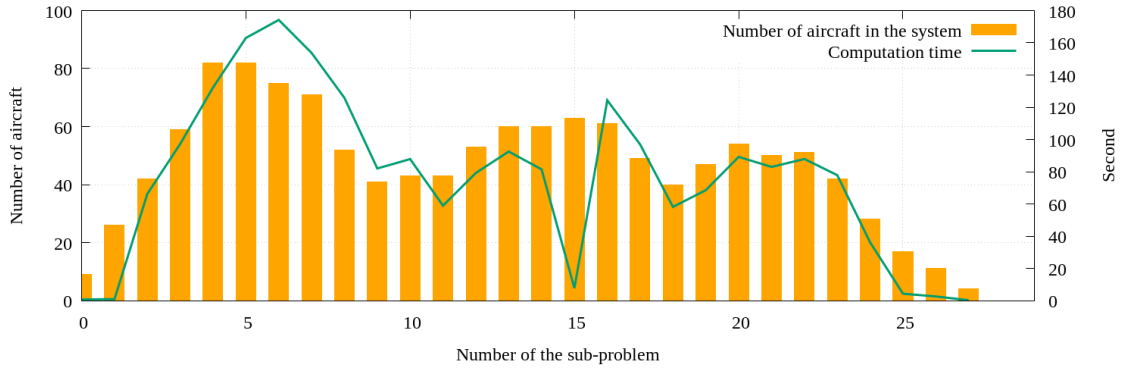


Figure 4.16: The computational time and number of aircraft in the system associated with the sub-problem.

Time transferring analysis

The main advantage of the concept of E-AMAN is the early planning of the arrival traffic before entering the congested terminal area. Through dynamic control procedures, E-AMAN shifts workload from TMA to en-route sectors to achieve the required time margins that can ensure a smooth and efficient operation in the TMA. Therefore, the time transfer between the enroute phase and the TMA can explicitly reveal the performance of the dynamic control model.

Since the TMA entry points separate the enroute segment and the TMA, so aircraft arrival times at the TMA are analyzed. A baseline measurement is conducted by performing an optimization in the enroute phase with an aim of eliminating congestion, resolution workload and reducing the speed deviation. In the baseline model, the original time information of flights entering the boundary is used as input. The enroute network is also used. Enroute decision variables, which are aircraft speeds on the links and the alternative route assignment in the dynamic optimization model are identically implemented including the constraints related to their ranges. The objective of the baseline model is to minimize the sum of the congestion, conflict resolution workload and the speed deviation from the nom-

inal speed of each aircraft. The speed deviation is considered based on the fact that under the enroute sector control, aircraft are expected to be operated in a fuel efficient manner without incurring any congestion and excess of the conflict resolution workload. For an aircraft, the speed deviation is defined as $v_f^{Opt} - v_{l_i}^{u_f}$, where v_f^{Opt} and $v_{l_i}^{u_f}$ represent the nominal enroute speed and the assigned control speed on link l_i of aircraft f, respectively.

The transfer time is then computed for each aircraft by using the time obtained from the dynamic optimization model to subtract the final TMA entry times obtained from the baseline model. Fig. 4.17 shows the transfer time of each aircraft, in which the negative values indicate earliness and the positive values indicate tardiness. Numerical details are given in Tab. 4.5, where the results are investigated with respect to the four TMA entry points. The maximum time of earliness and tardiness associated with the four TMA entry points are given, where the biggest values are shown in bold which indicate that the transferred times are limited to less than 6 minutes. Moreover, the average transferred times without considering earliness or tardiness are all less than 2 minutes.

It is clear that long-term situation awareness helps to set suitable time margins for aircraft, and early control can achieve the required TTG/TTL through speed adjustment and route changing. By transferring a proper amount of time for aircraft, efficiency can be guaranteed by providing a smooth operational environment between the enroute segment and the TMA while preserving or enhancing the safety.

Considering the results of conflict reduction and transferred time, the time required to have almost all the safety issues solved is achievable. Though the conflicts have not been eliminated, they have been limited to a minimum number which can be solved easily by applying extra strategies at tactical level. Compared to the reduced holding time, and the reduced workload for controllers in the approach phase, the conflict can be solved without strongly increasing the cost. To sum up, the proposed approach has been perfectly adapted to the E-AMAN and can provide up-to-date solutions for aircraft, taking into account the future situation, thereby enabling efficient and safe arrival management.

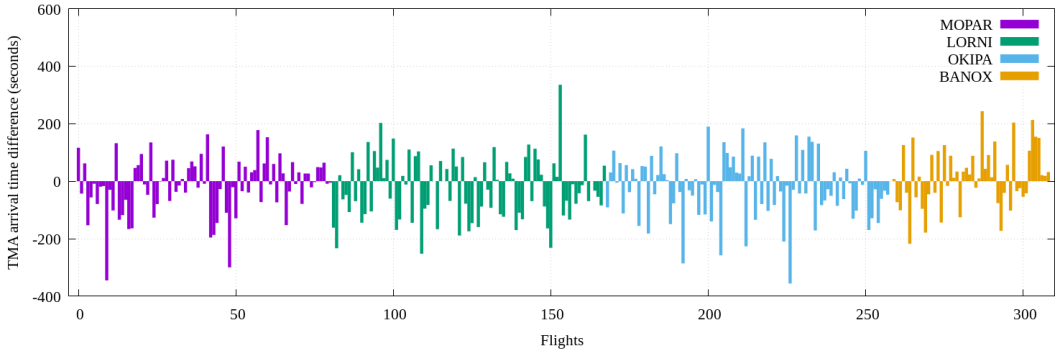


Figure 4.17: The gain or loss of time in the cruising phase by applying early trajectory adjustment.

4.5 Conclusion

In this paper, we investigate a dynamic control optimization problem based on the E-AMAN concept for arrival air traffic considering the main safety issues in the enroute phase and

Table 4.5: Time difference at TMA entry point between the schedule provided by the dynamic optimization model and the baseline case.

	Earliness (absolute value)			Tardiness (absolute value)		
	Nb of aircraft	Max (Sec)	Ave (Sec)	Nb of aircraft	Max (Sec)	Ave (Sec)
MOPAR	46	344.9	78.63	35	177.6	70.7
LORNI	49	252.3	102	38	**335.06**	79.8
OKIPA	51	**355.8**	92.8	40	189.65	72.1
BANOX	22	217.7	81.2	28	243.23	84.16

the TMA. E-AMAN concept relies on earlier trajectory adjustment under good situation awareness of the overall system. In line with this concept, we established a dynamic optimization model, aiming to provide a reliable decision support tool for arrival management.

The dynamic nature is addressed through the techniques of rolling horizon and freezing horizon. The rolling horizon is defined according to a 20-minutes information update cycle, while frozen horizon refers to a geographical area in which the decisions of aircraft are fixed. With the rolling horizon approach, the dynamic problem is tackled by periodically updating the aircraft trajectory information and updating the aircraft set according to whether the aircraft is in the system. Sub-problem associated with each information update incorporates the congestion, conflict resolution workload in the enroute phase and the conflicts in the TMA into the objective function. In order to enable a long term awareness of the air traffic situation, predictions of aircraft trajectories are made for the sub-problems. Additionally, considering the fact that as aircraft get closer to the TMA, the weight for the objective in the TMA should increase, we introduced a dynamic weighted approach. In this approach, for each aircraft, different weights are assigned for the enroute metrics and the TMA metrics with a sum value of 1 according to the position of the aircraft with respect to its route. Through this design, aircraft that are far away from the TMA are preferably controlled for en-route safety problems, while when aircraft approaches the actual TMA, more weights tend to be assigned to the metrics in the TMA.

Results of the static sub-problems are analyzed from two aspects. The evolution of the metrics, especially the conflicts in TMA implies that the proposed algorithm can effectively reduce safety issues through achievable time transfer. However, in a dynamic environment, when high traffic density is involved in the problem, the ability of early adjustment in the enroute phase may reach the limit, so still, there are few conflicts that need to be solved by applying other strategies.

For future work, the model will be improved to be more realistic by increasing the flexibility of decision assignments for aircraft during the cruising phase.

Chapter 5

An improved algorithm for arrival aircraft scheduling problem in the terminal area

Though it is proven that Simulated Annealing (SA) is efficient in solving problems, it is always of great value to seek for designs that improve computational efficiency, especially for the problems in this thesis, when timely responses to the new situations should be made. In a real operational environment, flights are independently controlled by adjusting operational parameters to ensure the required separations. Flight conflicts are detected between each pair of continuously operated flights, so changing the decisions of one flight does not have too much effect on the system. Considering this fact, the objective functions that involve the conflicts encountered by all flights in the previous chapters can be simplified. Therefore, we propose a single aircraft performance based objective function so as to avoid unnecessary evaluation and provide better information for performing guidance to high quality solutions.

The organization of this chapter is summarized as follows. Section 5.1 describes the problem that is addressed in this chapter. In Section 5.2, the mathematical model is introduced, in which the objective functions of the two models referred to as original algorithm and selective algorithm are given respectively. The details of applying SA to solve the problems corresponding to both models are described and presented in Section 5.3. Then, in Section 5.4, final results are analyzed and compared in terms of execution time and solution quality. Conclusions and perspectives are discussed in Section 5.5.

5.1 Problem statement

No matter whether the arrival air traffic is handled at a tactical level or online real-time level, a common interest is to improve the computational efficiency of the decision support tool. The arrival aircraft sequencing and scheduling problem that we focused on usually involves high traffic density. In addition, aircraft perform intensive maneuvers in a relatively small geographical area. Therefore, computational efficiency is of great importance to provide timely decision-making information for pilots and air traffic controllers in such a changeable environment.

Considering the fact that aircraft are organized under the commands from the controllers who get access to the information coming from other aircraft (current positions, speed, route direction, etc.), our problem can be addressed in accordance with individual performance. The algorithms proposed in the previous chapters compute the objective values by performing overall evaluation of all the flights in the network. While in the proposed algorithm, which is also referred to as selective algorithm, we select a set of flights that contribute the most to the objective value and use them to provide solution options. This process is expected to reduce redundant computation and provide better guidance for solution searching. The original algorithm and the selective algorithm lead to different neighborhood functions in the SA. Therefore, the solution algorithm are made corresponding adjustments according to the features of these two algorithms.

5.2 Mathematical models

This model is establish by referring to Ma *et al.* (2019). We focus on a simple aircraft scheduling problem in the Terminal Maneuvering Area (TMA) with a single runway case.

5.2.1 Network

Identical to the previous study in Chapter 3, the network is abstracted as a Graph $G(\mathcal{N}, \mathcal{L})$, where $\mathcal{N}$ represents the set of nodes, $\mathcal{L}$ represents the set of links. Each flight follows a designated route denoted as $\mathcal{U}_f = \{\mathcal{U}_f^l, \mathcal{U}_f^n\}$, where $\mathcal{U}_f^l$ and $\mathcal{U}_f^n$ represent the link set and node set that flight f passes through, respectively. The information of each flight ($f \in \mathcal{F}$) is specified:

- c_f: Wake turbulence category of flight f.
- e_f: TMA entry node of flight f.
- t_f^o: Initial Required Time of Arrival (RTA) when flight f entering the TMA through corresponding entry node.
- v_f^o: Initial speed of flight f when entering the TMA through the entry node.

5.2.2 Decision variables

In this problem, two types of decision variables are introduced: TMA entry time t_f and TMA entry speed v_f for flight f, which are chosen in a discretized manner identical to Chapter 3:

$$t_f \in T_f := \{t_f^o + j\Delta t | j \in \mathbb{Z} \text{ and } t_f^{\min} \leq j\Delta t \leq t_f^{\max}\}$$

$$v_f \in V_f := \{v_f^o + j\Delta_f^v | j \in \mathbb{Z} \text{ and } v_f^{\min} \leq j\Delta_f^v \leq v_f^{\max}\}$$

where T_f and V_f are the sets of available values of the TMA entry time and TMA entry speed of aircraft f. Δt and Δ_f^v are the time slot and speed slot for selecting the values of decision variables. Our decision vector is represented by $\mathbf{x} = (\mathbf{t}, \mathbf{v})$, where $\mathbf{t}$ is the TMA

entry time vector and v is the TMA entry speed vector. For a flight f, its decision vector can be represented as $\mathbf{x}_f = (t_f, v_f)$.

5.2.3 Conflict detection

The conflict detection is categorized into two types due to the separation requirements. Link conflict detection includes evaluating the wake turbulence separation violation and overtaking conflict. Node conflict detection is introduced for the horizontal separation violation.

Link conflict detection

As shown in Fig. 5.1, for each link $l = (\nu_1, \nu_2)$, where ν_1 and ν_2 represent the entry and exit point of link l, respectively. The detection of wake turbulence separation violation is carried out on both sides of the link. In order to evaluate the separation violation severity, the percentage of violated distance over the required separation is added to the measurement of the conflict. The violation severity can provide extra information for optimization process. Suppose that two aircraft f and g pass through link l, where aircraft f is ahead of g. For f and g, $C^l_{f,g}(\mathbf{x})$ represents the conflict related cost on link l. It takes into account the conflict detection and evaluation on both the entry point and the exit point of link l. Therefore, we have:

$$C^l_{f,g}(\mathbf{x}) = C^{l_{in}}_{f,g}(\mathbf{x}) + C^{l_{out}}_{f,g}(\mathbf{x}) \tag{5.1}$$

where $C^{l_{in}}_{f,g}(\mathbf{x})$ and $C^{l_{out}}_{f,g}(\mathbf{x})$ represent the cost of aircraft f and g at the entry point and exit point of link l, respectively. Specifically, they are given as:

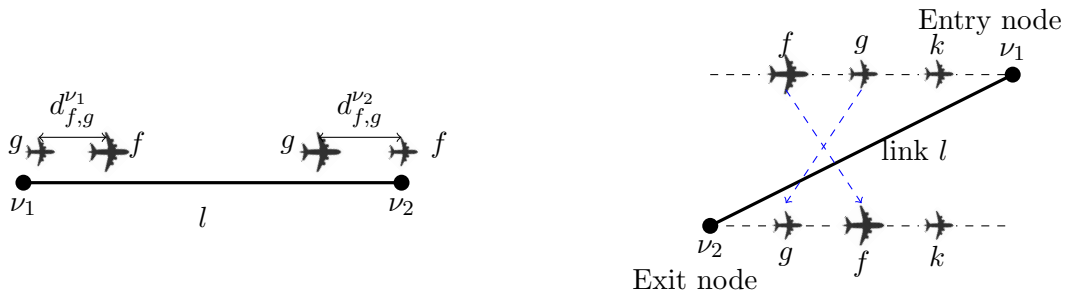
$$C^{l_{in}}_{f,g}(\mathbf{x}) = \begin{cases} 1 + \frac{s_{f,g} - d^{\nu_1}_{f,g}}{s_{f,g}}, \text{ if } d^{\nu_1}_{f,g} < s_{f,g} \\ 0, \text{ otherwise} \end{cases} \tag{5.2}$$

$$C^{l_{out}}_{f,g}(\mathbf{x}) = \begin{cases} 1 + \frac{s_{f,g} - d^{\nu_2}_{f,g}}{s_{f,g}}, \text{ if } d^{\nu_2}_{f,g} < s_{f,g} \\ 0, \text{ otherwise} \end{cases} \tag{5.3}$$

where $d^{\nu_1}_{f,g}$ and $d^{\nu_2}_{f,g}$ are the separations between aircraft f and g when aircraft g arrives at the link entry point and aircraft f arrives at the link exit point of link l. $s_{f,g}$ is the required wake turbulence separation associated with the categories of aircraft f and g which are given in Tab. 5.1.

Aircraft follow exactly the assigned routes, overtaking conflicts between two consecutively operated flights is then considered. In this case, overtaking conflict is evaluated on each link by comparing the sequence order of each aircraft when it arrives at the entry point and at the exit point of this link. For an aircraft f, the overtaking conflict $C^{l_{ov}}_f(\mathbf{x})$ on link $l = \{\nu_1, \nu_2\}$ is quantified by:

$$C^{l_{ov}}_f(\mathbf{x}) = \sum_{\substack{l \in \mathcal{U}^l_f \\ l=(\nu_1,\nu_2)}} |O^{l_{\nu_1}}_f - O^{l_{\nu_2}}_f| \tag{5.4}$$



(a) Wake turbulence separation violation detection.

(b) Overtaking conflict detection on link.

Figure 5.1: Link conflict detection configuration includes the detection of wake turbulence separation violation at link entry point and link exit point, and the overtaking conflict detection.

where $O_f^{l_{\nu_1}}$ and $O_f^{l_{\nu_2}}$ represent the sequence order when flight f gets in and gets out of the link l among all flights. In Fig. 5.1b, an example is illustrated, in which three aircraft k, f, g pass the link $l = \{\nu_1, \nu_2\}$ successively. The difference in the sequence order of f and g when they pass ν_1 and ν_2 indicates that an overtake has occurred. Therefore, the overtaking conflict is assigned to both f and g.

Node conflict detection

Due to the fact that the horizontal separation requirement of flights has risks of being violated around the joint of two links, each node is considered as a disk with a radius of 2.2NM. The distance of 2.2NM is computed based on the horizontal separation standard, aircraft speeds and the constructed angles of arrival routes. Computation details can refer to Ma *et al.* (2016). This area is defined as the *detection zone*. If two aircraft were observed in the detection zone simultaneously, a node conflict would occur. The violation severity on the node is computed by taking into account the proportion of the violation time to the smaller flying time in the detection zone of the two aircraft. Fig. 5.2 shows the node conflict detection configuration where two consecutively operated aircraft f and g pass through the detection zone of node n, where f is ahead of g. The node conflict cost $C^n_{f,g}(\mathbf{x})$ for aircraft f and g on node n is given as:

$$C^n_{f,g}(\mathbf{x}) = \begin{cases} 1 + \frac{t^{Out}_{f,n} - t^{In}_{g,n}}{\min(T^n_f, T^n_g)}, & \text{if } t^{Out}_{f,n} > t^{In}_{g,n} \\ 0, & \text{otherwise} \end{cases} \tag{5.5}$$

Table 5.1: Minimum wake turbulence separation requirements, in NM.

Categories		Leading Aircraft, f		
		Heavy	Medium	Light
Trailing Aircraft, g	Heavy	4	3	3
	Medium	5	3	3
	Light	6	5	3

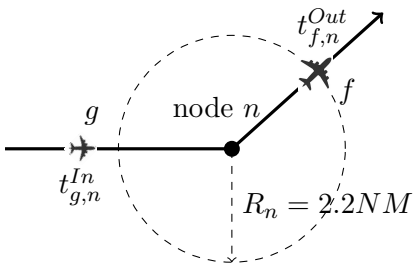


Figure 5.2: Node conflict detection configuration. If aircraft f and g were operated in the detection zone simultaneously, a conflict would be detected.

where $t^{Out}_{f,n}$ and $t^{In}_{g,n}$ denote the exit time of aircraft f and entry time of aircraft g at the detection zone of node n, respectively. T^n_f and T^n_g are the flying time of aircraft f and g when they pass through the detection zone of node n.

5.2.4 Delay and speed deviation

In order to provide a comprehensive analysis on the performance of the algorithms, other practical issues that need to be considered are included. In real operations, delay and speed deviations are two factors need to be minimized. Therefore, they are incorporated into our evaluation.

Flight delay D_f is computed as the time deviation between the assigned TMA entry time and the required one t^o_f. Thus for flight f we have:

$$D_f(\mathbf{x}) = |t_f - t^o_f| \tag{5.6}$$

The minimization of speed deviation can produce fewer changes and ease the workload for both the pilots and the air traffic controllers. The speed deviation V_f of flight f is given by the following equation:

$$V_f(\mathbf{x}) = \frac{|v_f - v^o_f|}{v^o_f} \tag{5.7}$$

where v^o_f is the initial TMA entry speed of flight f.

5.2.5 Objectives

Two objective functions are presented. One of them is formulated in an original manner and the other one is formulated based on the single aircraft performance. The original model and selective model are therefore established.

Objective function of the original model

In the original model, the objective function is a weighted sum of the total number of conflicts in the network, the total delay and speed deviations of all flight. It is given as

follows:

$$G_C(\mathbf{x}) = \alpha \left\{ \sum_{\substack{f,g \in \mathcal{F} \\ f \neq g}} \left(\sum_{l \in \mathcal{U}_f^l \cap \mathcal{U}_g^l} C_{f,g}^l + \sum_{n \in \mathcal{U}_f^n \cap \mathcal{U}_g^n} C_{f,g}^n \right) + \sum_{f \in \mathcal{F}} \sum_{l \in \mathcal{U}_f^l} C_f^{l_{ov}} \right\} + \beta \sum_{f \in \mathcal{F}} (D_f + V_f) \quad (5.8)$$

In our problem, safety is the most important issue that needs to be minimized with priority. Thus α is set much bigger than β.

Objective function of the selective model

The selective model identifies the performance of each flight in order to build an individual based objective function. For a given flight f, its weighted sum of the conflict cost, delay and speed deviation is computed as the total cost of this flight, which is given as follows:

$$c_f(\mathbf{x}) = \alpha \left(\sum_{\substack{g \in \mathcal{F} \\ g \neq f}} \sum_{l \in \mathcal{U}_f^l \cap \mathcal{U}_g^l} C_{f,g}^l + \sum_{l \in \mathcal{U}_f^l} C_f^{l_{ov}} + \sum_{\substack{g \in \mathcal{F} \\ g \neq f}} \sum_{n \in \mathcal{U}_f^n \cap \mathcal{U}_g^n} C_{f,g}^n \right) + \beta \left(D_f + V_f \right) \quad (5.9)$$

In order to achieve an overall optimization, the most expensive individual cost among flights is the upper bound of all costs. This value is considered as the objective and will be minimized. By reducing the upper bound cost, the performance of all flights can be optimized. For the selective model, the objective function is given as follows:

$$G_D(\mathbf{x}) = \max_{f \in \mathcal{F}} (c_f(\mathbf{x})) \quad (5.10)$$

5.3 Implementation of simulation annealing

In Chapter 2, we have described the details about how SA works to solve the problem. The general procedures of SA are not described here. We elaborate the adapted SA that corresponds to each model, and the two versions of SA are referred to as Original Simulated Annealing (OSA) and Selective Simulated Annealing (SSA). It is worth mentioning that our optimization process is based on aircraft trajectory simulation. Therefore, the decision changes of a flight lead to new trajectory information of this particular flight. This provides a condition for focusing on the performance of one single flight for the problem solving of the selective model. Differences between the two algorithms mainly lie in the neighborhood selection and the solution evaluation, since those are two key steps that affect the efficiency of an algorithm. The details of both original and selective SA are specified in the following.

5.3.1 Original Simulated Annealing (OSA)

Heating procedure of OSA

Opposite to the cooling procedure, the temperature in the heating loop starts from a small value. Then, this value increases progressively by multiplying a control parameter δ that is slightly greater than 1. Each iteration corresponds to a temperature, and at each temperature, multiple transitions are executed. In each transition, steps of neighborhood solution

generation, solutions comparison and solution acceptance or rejection are performed. We define an acceptance rate ζ, which is computed by the number of neighborhood solutions accepted divided by the total number of transitions. As the total number of transitions increases, the acceptance rate ζ increases as well. Once the acceptance rate reaches a certain value, typically 0.8, the heating process stops and the current temperature is used as the initial temperature for the cooling procedure.

Cooling procedure of OSA

As we use SA for problem solving, the procedures such as temperatures, neighborhood selection and acceptance probability are identically applied. In the OSA, the neighborhood solution is chosen by applying the roulette wheel selection approach. This approach ensures a random selection of a flight in the flight set with a chance associated with the flight cost. In this approach, the total cost of all flights makes up a roulette wheel, where the cost of each flight is proportional to the likelihood of being selected. A new solution is generated and the associated objective value can be obtained.

In order to evaluate the quality of the neighborhood solution, the information associated with the chosen flight needs to be removed from the simulation environment. After updating the decision variable of this flight, its trajectory with new decisions is simulated. Then, an overall evaluation for all flights is performed, from which the objective value of this neighborhood solution is then yielded. The acceptance rate of the neighborhood solution is dependent on the metropolis criterion which can be seen in Chapter 2.

Two potential drawbacks in the OSA in terms of computational efficiency can be noticed. In this case, the objective function integrates the information of all flights. Therefore, first, the selection of neighborhood solutions is rather blind. Second, evaluating the neighborhood solution requires a lot of calculations.

5.3.2 Selective Simulated Annealing (SSA)

The SSA focuses on critical individuals which are essential in coping with undesired situations. The heating procedure and the cooling procedure are thoroughly introduced in terms of two main different steps: the neighborhood solution and the solution evaluation.

Heating procedure

In the heating procedure of the SSA, the performance of each flight is considered. For each iteration, the flights are successively chosen according to their order in the flight set. Taking one flight as example, in a random transition, flight f is chosen, whose neighborhood solution $\mathbf{x}'_f$ is generated by changing one of its decision variables. The costs associated with f before and after the decision change are represented as $c_f(\mathbf{x}_f)$ and $c_f(\mathbf{x'}_f)$. The acceptance of this neighborhood solution for flight f is partially determined by the cost of current solution and the cost of the neighborhood. The acceptance condition can be found in Chapter 2. Similar to the OSA, temperature increases when all flights finish repeating the aforementioned procedure. Once the acceptance rate reaches a certain value (typically 0.8), the heating process terminates.

Cooling procedure

In the cooling process of the SSA, one important step is to find out the group of flights with the highest level of costs, namely the critical flight set. For each iteration, the overall status of the system is initiated. All flights are simulated and their associated costs are obtained. Among all costs, the highest one denoted as C_m is specified as a reference for defining cost threshold. Then, the cost of each flight is compared with the cost threshold calculated by λC_m, where λ is a threshold parameter. Consequently, the flight whose cost is higher than the cost threshold is put into the critical flight set $\mathcal{F}_c$. This process is summarized in Fig. 5.3, where the flight set is listed on the x-axis with their associated costs indicated on the y-axis. In this example, the maximum value of cost in this iteration is 19.71. Thus the critical flight set is composed of the flights with costs higher than $19.71 \cdot \lambda$. The horizontal red line represents the value of cost threshold used for filtering critical flights. Flights with their costs marked by red circles have their costs higher than the cost threshold, and therefore are classified as critical flights in this iteration. After this process, flights in $\mathcal{F}_c$ are selected successively at each transition, for which the selected flight generates a neighborhood solution by randomly changing one of its decision variables. Optimization is performed by focusing on the cost of specific flights with the following steps. First, evaluate the cost of this flight after the decision change. Second, compare the costs of this flight before and after the decision change. Third, decide whether to accept this decision change. The whole optimization process of the SA including neighborhood selection method is displayed in Algorithm 6. Through targeting the critical flights one by one, the algorithm is expected to be efficient in terms of information volume and neighborhood generation time. An overall evaluation of all flights is regularly performed but much less often than in the OSA.

Fig. 5.4 and Fig. 5.5 illustrate the algorithm structures of OSA and SSA, in which the differences reflect in two aspects. On one hand, in the OSA, there is no strong preference

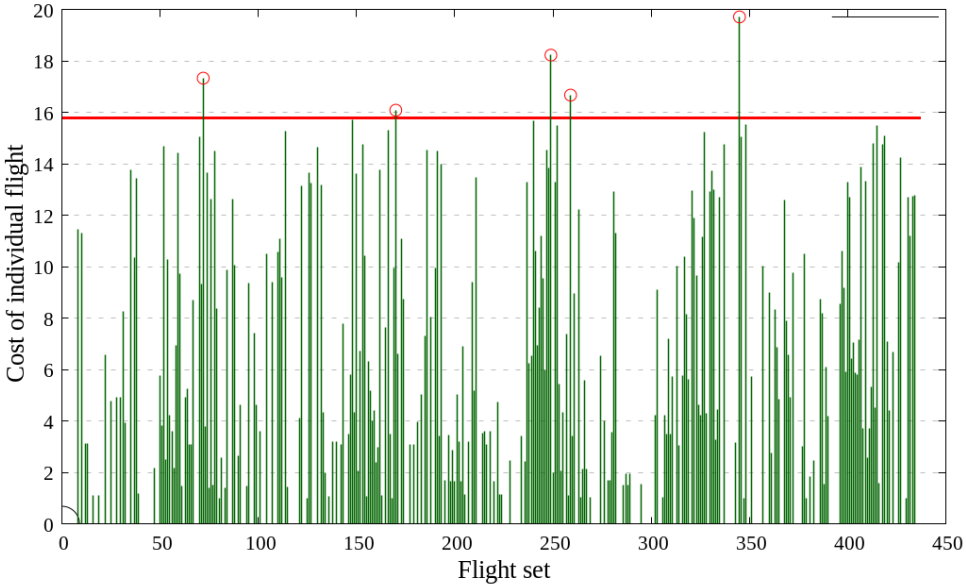


Figure 5.3: Critical flights selection process of the SSA.

Table 5.2: Chosen parameter values for solution algorithms.

Parameters of the solution approaches	Values
Heating control parameter for OSA and SSA, δ	1.1
Heating acceptance rate for OSA and SSA, ζ	0.8
Cooling parameter for OSA, γ_c	0.99
Number of transitions in each iteration for OSA, I	100
Cost threshold factor for SSA, λ	0.8
Cooling parameter for SSA, γ_d	0.999
SA stopping criterion for OSA	$0.00001T_o$
SA stopping criterion for SSA	$0.0001T_o$

for neighborhood selection, so the state space is randomly explored, which is inefficient in finding good solutions. However, in the SSA, the cost of each flight is considered as reference for the neighborhood selection process, in which flights with high costs are prioritized to be selected for performing the decision change. This process provides guidance for neighborhood selections, therefore better solutions can be found much easier and faster.

Algorithm 6 Details of the SSA, including the neighborhood selection method.

Initialize:
    Initial temperature T_o obtained from heating procedure, random number $\beta \in [0, 1]$;
for $f \in \mathcal{F}$ **do**
    Compute the cost $c_f(\mathbf{x}_f)$ of flight f;
end for
Specify the maximum individual cost C_m of all flights;
while $T_c > 0.0001 \cdot T_o$, $C_m > 0$ **do**
    for $f \in \mathcal{F}$ **do**
        if $c_f(\mathbf{x}_f) \geqslant \lambda C_m$ **then**
            Put flight f in the critical flight set $\mathcal{F}_c$;
        end if
    end for
    for $f \in \mathcal{F}_c$ **do**
        One of decision variables (t_f, v_f) is chosen with equal probability and change its value.
        Calculate the new cost of f, $c_f(\mathbf{x}'_f)$.
        if $c_f(\mathbf{x}_f) > c_f(\mathbf{x}'_f)$ **then**
            $\mathbf{x}_f \leftarrow \mathbf{x}'_f, \quad c_f(\mathbf{x}_f) \leftarrow c_f(\mathbf{x}'_f)$;
        else if $\beta < \exp(\frac{c_f(\mathbf{x}_f) - c_f(\mathbf{x}'_f)}{T_c})$ **then**
            $\mathbf{x}_f \leftarrow \mathbf{x}'_f, \quad c_f(\mathbf{x}_f) \leftarrow c_f(\mathbf{x}'_f)$;
        end if
    end for
    $T_c = T_c \cdot \gamma_d$;
    for $f \in \mathcal{F}$ **do**
        Update the cost $c_f(\mathbf{x}_f)$ of flight f;
    end for
     Specify the maximum individual cost C_m of all flights;
end while

On the other hand, the objective value of the original model requires an evaluation involving all flight information, while in the selective model, performance of a single flight is focused on the objective function. Therefore, the solution evaluation requires the trajectory information of a small number of flights. Consequently, computational time for solution evaluation is reduced. Mention that in Fig. 5.5, the subscript f is omitted in each notation of the decision variables in order to improve simplicity. In Tab. 5.2, the user-defined parameters for OSA and SSA are given.

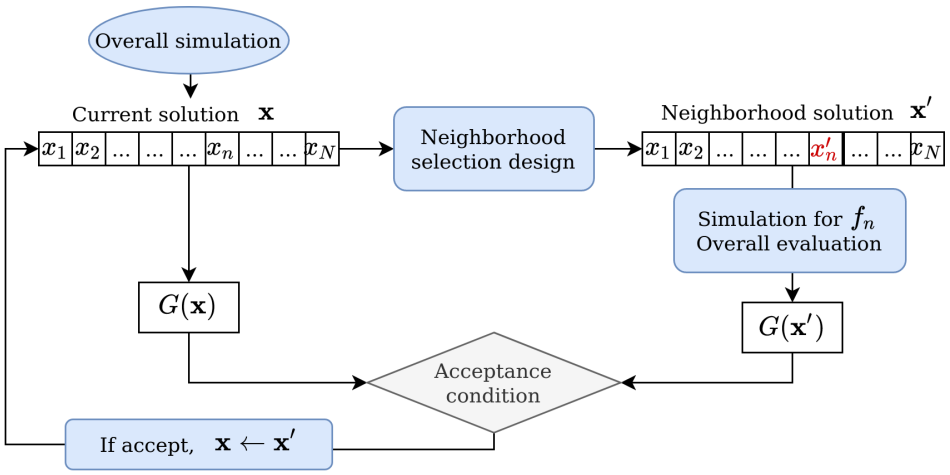


Figure 5.4: Algorithm structure of OSA, in which the information of all flights is considered for each evaluation.

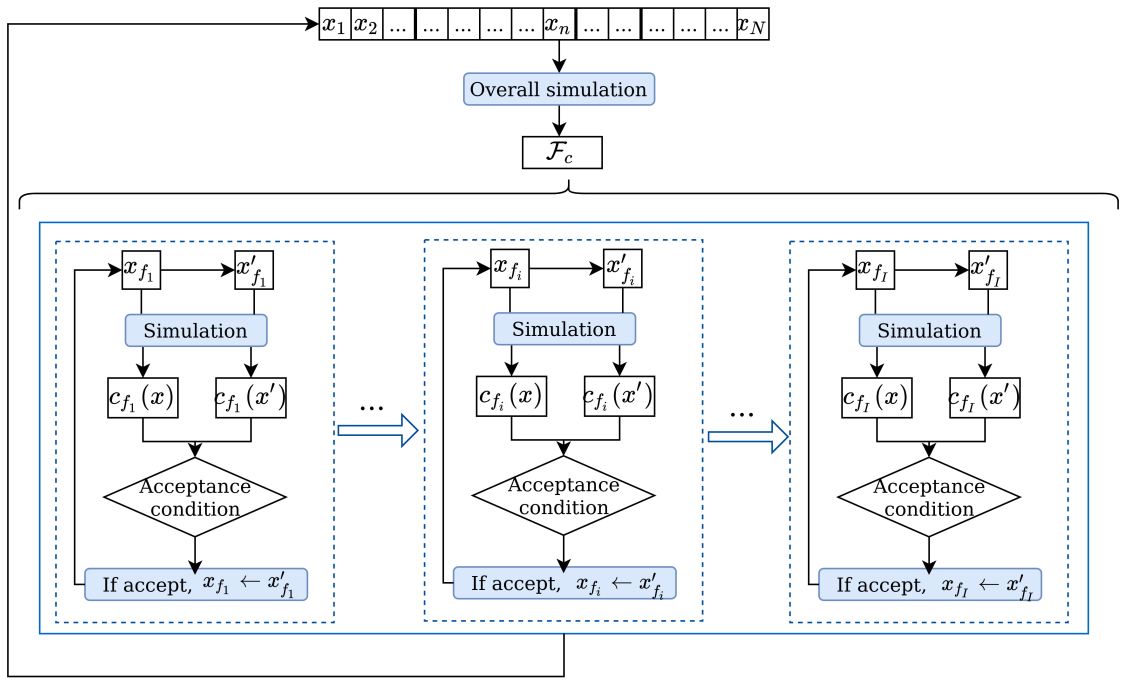


Figure 5.5: Algorithm structure of the SSA. Features of the selective model are emphasized by considering the performance of a single flight at each transition.

5.4 Results

5.4.1 Case study

Actual arrival flights landing on runway 26L on July 11th 2017 at the Paris CDG airport are applied for our case study. Table. 5.3 provides the flight information by associating the mixed ratio of aircraft categories and the TMA entry point. Flights are mostly composed of heavy and medium aircraft. Fig. 5.6 depicts the arrival route network of the Paris Charles de Gaulle (CDG) airport. In our case study, the parameters chosen for the model are given in Tab. 5.4. It is worth mentioning that $\Delta T_{\max}$ is set to 30 mins which is a large range. Since delay and speed deviation are included in the objective function, we want to provide a large enough margin to make sure a conflict free solution can be achieved. On this basis, the optimization of delay and speed deviation is reasonable. In addition, tardiness of 30 mins seems unreal for short-haul flights as a result of the speed regulation. If a high tardiness time is required, the ground delay can be considered as a time control measurement. The overall process is run on a 2.50GHz core i7 CPU, under Linux operating system PC with Java code. The derived results are investigated in terms of execution time and the solution quality for the two models.

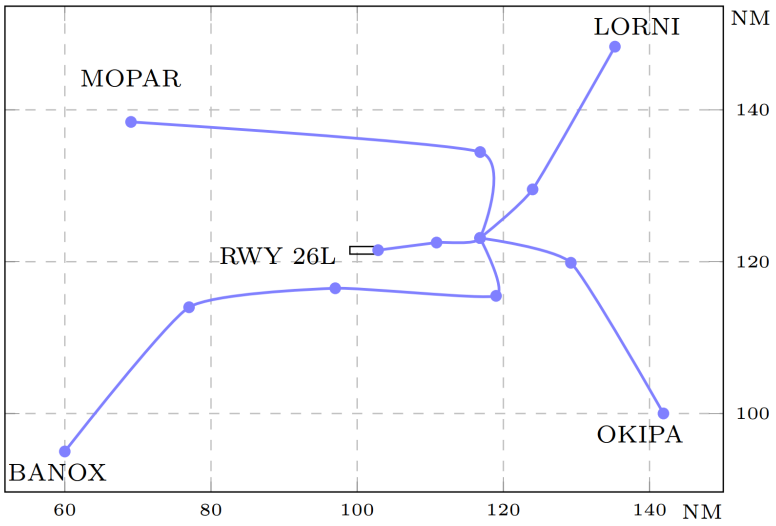


Figure 5.6: Arrival route structure for runway 26L of the Paris CDG airport.

Table 5.3: Arrival flights information of the heavy and medium flights with respect to the TMA entry node.

Entry point	Total number of flight	Heavy	Medium
MOPAR	60	26	34
LORNI	74	20	54
OKIPA	195	36	159
BANOX	108	20	88
Total number	437	102	335

Table 5.4: Chosen parameter values for the optimization problem

Parameters of the optimization problem	Values
Maximum tardiness regarding RTA, $t_f^{\max}$	30 mins
Maximum earliness regarding RTA, $t_f^{\min}$	-5 mins
Time slot, Δt	5 seconds
Maximum speed in TMA, $v_f^{\max}$	$1.1v_f^o$
Minimum speed in TMA, $v_f^{\min}$	$0.9v_f^o$
Speed slot, Δ_f^v	$0.01v_f^o$
Weighting parameter, α	1
Weighting parameter, β	0.001

5.4.2 Execution time

In the SA, the main factor that affects the execution time of the optimization process is the time needed for the neighborhood selection and solution evaluation. In order to accurately measure the performance of the OSA and the SSA in terms of the execution time, a metric that approximates the total runtime is defined for each temperature (iteration) by the following equation:

$$TO = \sum_{k=1}^{K}(TN + TE) + L \tag{5.11}$$

where TO denotes the total computation time for a particular iteration. TN represents the time for generating a neighborhood solution and TE is the sum of the time needed for evaluating the cost of the neighborhood solution. K is the total number of transitions required for this iteration and L is the time needed to transfer from one iteration to the next one. To distinguish the parameters for the two algorithms, we add the subscript of OSA and SSA to the associated parameters for the following representations. The execution times of the two algorithms are compared according to each parameter shown in Eq. 5.11.

One important factor that affects the computation time of an iteration is the number of transitions in the iteration. In the SSA, at the beginning of each iteration, an overall evaluation should be done to specify the costs of all flights, based on which, the value of C_m is updated for the current iteration. Since different values of C_m lead to different threshold values for the selection of critical flights, the number of transitions which is determined by the flight number in $\mathcal{F}_c$ varies for each iteration. Considering the threshold parameter $\lambda = 0.8$, it is expected that only a small proportion of flights belongs to the critical flight set. However, in the OSA, the number of transitions for each iteration K_{OSA} is fixed as 100. Therefore, normally K_{OSA} is higher than K_{SSA}.

TN is dependant on the neighborhood selection methods. The roulette wheel selection method used by OSA includes several steps such as generating a random number, adding the cost of each flight successively to the partial sum, checking whether the condition of selecting the flight is satisfied or not. In this approach, the time spent in searching for the flight that performs a decision change is dependent on the value of the random number. If the random number is small, the time for generating a neighborhood solution is probably

short. In this case, random selections introduce unstable values of the TN. Whereas in the SSA, the flights that perform decision changes are already specified in the critical flight set $\mathcal{F}_c$. Compared to the OSA, the time spent for random flight searching process is saved. As a consequence, TN_{OSA} is higher than TN_{SSA}.

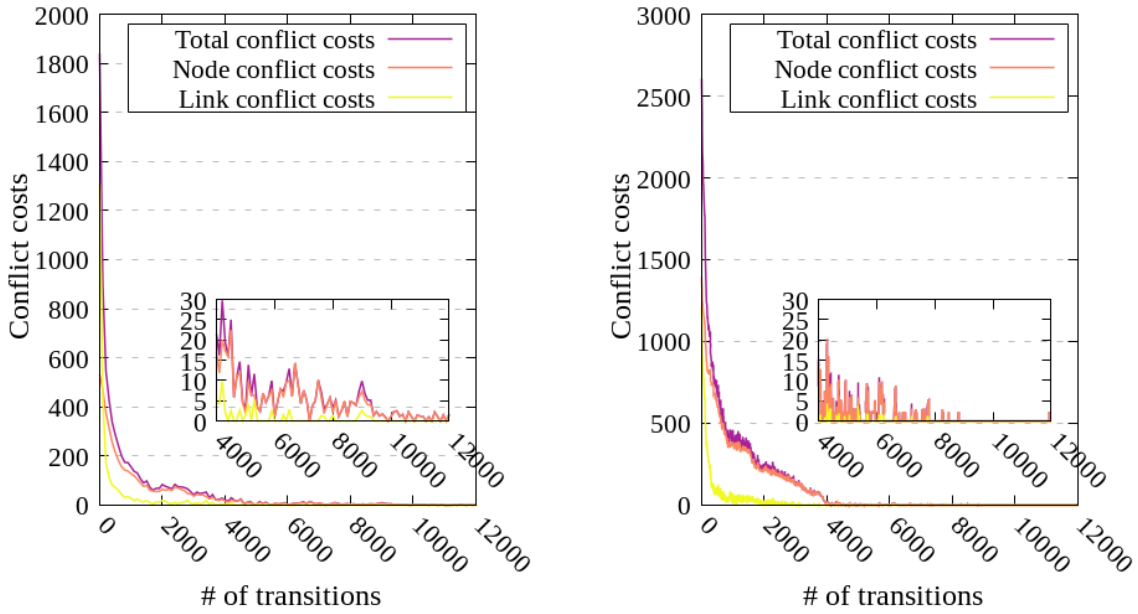
For the TE, the execution time for neighborhood evaluation involves trajectory prediction. This part is identically carried out in the OSA and the SSA. Apart from these, TE also includes the execution time of the solution evaluation. As previously defined, in the OSA, the costs of conflicts, delays and speed deviations are evaluated for each pair of flights on the associated resources and then summed up, namely, 437 flights in this problem are all involved. While in the SSA, only the chosen flight is focused, and its cost can be obtained by checking the interactions with adjacent flights that pass the same resources. In this case, irrelevant interactions between other flights are not considered, which ensures that the SSA is much faster than the OSA in terms of objective function evaluation. Considering the above analysis, consequently, corresponding to each k, TE_{SSA} is smaller than TE_{OSA}.

L_{SSA} is slightly higher than L_{OSA} because it includes an overall flight cost evaluation to initiate the maximum individual cost. However, the time of the overall evaluation in L_{SSA} can be easily offset by the solution evaluation of any transition in the OSA. By analysing the elements in Eq. 5.11, we can conclude that TO_{SSA} is much smaller than TO_{OSA}.

The number of iterations also plays an important role in the total computation time. As each iteration corresponds to a temperature, the number of iterations is related to three parameters: the initial temperature obtained from the heating procedure, the cooling parameter, and the stopping criterion. Tab. 5.5 gives the execution time information of the OSA and the SSA. We can notice a significant difference in the computation time of the OSA and computation time of the SSA. The SSA shows great advantage in improving the computational efficiency as it is much more faster than the OSA. In the table, the number of iterations and transitions related to the optimization process are given. We can see that the number of iterations of the SSA is larger than those of the OSA. In comparison, the total number of transitions is the opposite. This is because in the SSA, the number of transitions is determined by the number of critical flights, which is changing for each iteration. It is worth mentioning that from the initial consideration, the computation time of each transition is summed up successively according to Eq. 5.11. However, though in our study, SSA is conducted based on single thread computation, it allows parallel execution for multiple flights when multiple processors are available, which can further improve the computational efficiency.

Table 5.5: Execution times of the OSA and the SSA.

	selective SA	Original SA
Number of iterations	9227	1146
Number of transitions	32298	114600
Total execution time(s)	42.869	1731.859
Average time for each transition(s)	0.00133	0.01511



(a) Conflict reduction of the OSA.

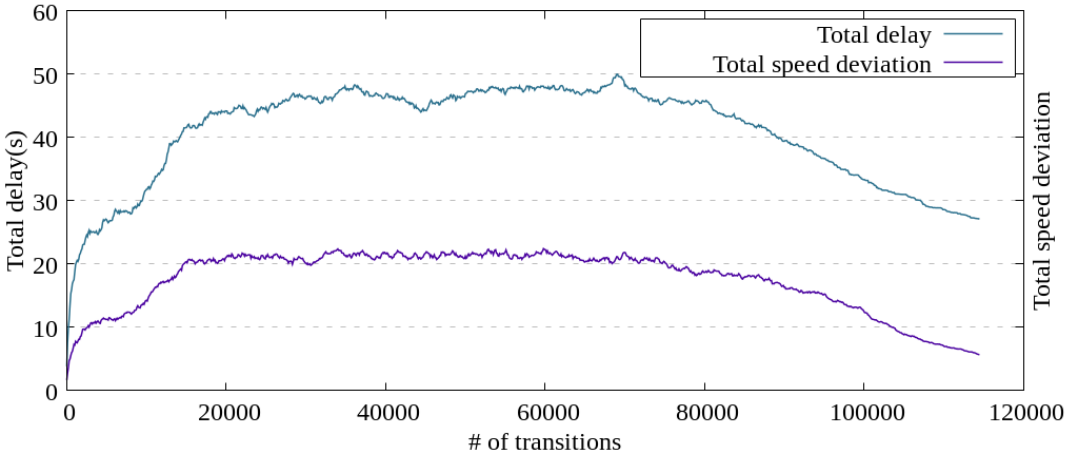
(b) Conflict reduction of the SSA.

Figure 5.7: Conflict evolution of the OSA and SSA during the optimization.

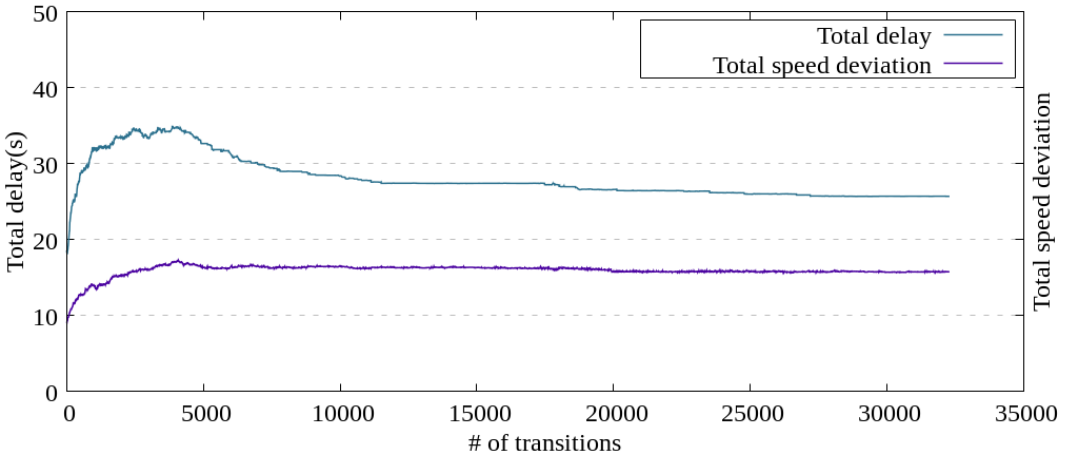
5.4.3 Analysis of the results

The qualities of the solutions of both algorithms are analyzed in terms of the components included in the objective function. Fig. 5.7 depicts the optimization process in terms of conflict reduction of the OSA and the SSA respectively. The conflicts have been eliminated in both cases. The figures have limited the x-axis to 12000 transitions, after which the numbers of conflicts stay close to 0. In order to clearly show the fluctuation of the numbers of conflicts when the values are small, a magnified window is added in each of the main figure. From the small figures, we can observe that the SSA have eliminated the conflicts earlier than the OSA by comparing the number of transitions required in both cases. This means that targeting the flights with high costs indeed improves the computation efficiency by providing proper guidance for the algorithm.

The optimization processes reflected on delay and speed deviation with respect to the number of transitions are displayed in Fig. 5.8. In order to maintain consistency and obtain a fair comparison of the two algorithms, the sum of the delay and the sum of the speed deviation of all flights are displayed without considering the weights used for the objective function. In Fig. 5.8a and Fig. 5.8b we can observe that the shapes of the graphs that represent the values of delay and speed deviation in each figure are consistent. However, for the two algorithms, different neighborhood functions lead to different state space exploration manners, therefore the variations of delay and speed deviation in the two models have different patterns. In Fig. 5.8a, the increase of delay and speed deviation continues for more than half of the optimization time, then the values decrease gradually. While in Fig. 5.8b, the values have a brief rise and fall, and then flatten out. It is known that the weight of the conflict is far bigger than the weight of the delay and the speed deviation in the objective function. Then it is normal to solve the conflict issues first. This leads to the increases in delay and speed deviation. From the figures indicating conflict



(a) Delay and speed deviation of the OSA.



(b) Delay and speed deviation of the SSA.

Figure 5.8: Delay and speed deviation evolution of the OSA and the SSA.

evolution (Fig. 5.7a and Fig. 5.7b), conflicts have been almost eliminated after around 5000 transitions. Observing the variations of delay and speed deviation in Fig. 5.8a and Fig. 5.8b after 5000 transitions, the OSA explores the state space freely and blindly, which makes it hard to find better solutions when there are some metrics associated with low weight. Whereas, the SSA is very effective for finding better solutions by accurately specifying the flights that cause the issues, therefore it can continuously generate better solutions which expedites the optimisation process. However, as more transitions finish, the SSA limits the neighborhood selection on a particular set of flights. If the decision changes of flights in the critical set do not provide better solutions, then critical flights always remain unchanged, which in turn leads to a situation that is not able to be optimized and hard to break. This fact reflects the drawback of the SSA. The final results in Tab. 5.6 show that under the current setting of parameters, the OSA provides a result with almost the same delay time as the SSA, but less speed deviation. The longer execution time certainly helps the optimization of the results of OSA. For the drawback of SSA, we could change the weight factors of the objective functions of the SSA or introduce some randomness in the critical flight selection to improve its performance. It is worth mentioning that Tab. 5.6

Table 5.6: Final values of delay and speed deviations for the OSA and the SSA.

	SSA	OSA
Delay (hour)	25.696	27.064
Speed deviation (%)	15.76	5.68

gives the cumulative delay and speed deviation for all the flights and we can expect a small value for a single flight. To summarize, the SSA shows very good performance in improving computational efficiency but the quality of solution is slightly worse than the OSA.

5.5 Conclusion

In this chapter, we focus on the improvement of computational efficiency of the solution algorithm. The original model is reconsidered and we found that unnecessary computation can be reduced by focusing on the performance of a single aircraft. Then a model with a rather isolated structure is proposed by adjusting the objective function and this model is referred to as the selective model. In order to further evaluate the performance of the solution quality, the delay and speed deviation are also included in the objective function. Instead of evaluating the total cost of all flights, we chose to minimize the maximum cost associated with a single flight. Once the maximum cost of an individual flight has been reduced, the whole system has been optimized. SA is applied to solve the problem corresponding to the proposed model and the originally formulated model. Optimization details have been specified. The most important differences lie in the neighborhood selection and the objective function evaluation.

The full-day data of the Paris CDG airport on one landing runway has been applied as case study. The results of the two models have been analyzed and compared in terms of execution time and quality of the result. Results show that the selective algorithm has an absolute advantage over the original one in execution time. The solution quality, however, is slightly worse than the original algorithm. A potential improvement for SSA could be achieved by adjusting the weight factors in its objective function or incorporating randomness in the flight selection process of the neighborhood function. Considering the number of flights that share the delay, the solution quality has no significant difference. By analyzing the optimization pattern combined with the structural features of the two algorithms, the potential advantages and drawbacks of each algorithm are described.

For the future work, we want to provide more tests with different scenarios and conduct a deep investigation on the attributes of the selective algorithm. We hope that the algorithm can be applied to the on-line support tool. The proposed algorithm shows great adaptability and flexibility, and it is possible to be applied for solving problem related to Unmanned Traffic Management (UTM).

Chapter 6

Conclusion and perspectives

This chapter summarizes the main contributions of this thesis and the perspectives. Based on this thesis, several promising research directions are envisaged, which provide a clear vision for the following work. Section 6.1 summarizes this thesis, then Section 6.2 proposes several research directions for the future works. Section 6.3 lists the publications resulting from this thesis.

6.1 Summary of this thesis

In this thesis, we presented decision support methodologies associated with Arrival MANagement (AMAN) in different geographical horizons while considering the uncertainty arising during flight operation. The first problem focuses on the scheduling of aircraft in the Terminal Maneuvering Area (TMA). In this problem, if aircraft need to adjust its time to enter TMA, common procedures would be stacking and holding, which will introduce fuel consumption, emission, noise effects and extra workload for air traffic controllers and pilots. Then, in the second problem, the arrival management horizon has been extended into the cruising phase which is up to 500NM from the destined airport to conduct early sequencing and collaborative enroute control to adjust the aircraft arrival time at the TMA. In our study, we believe that the prior goal of AMAN is to ensure safety. Therefore, in all of our problems, the detection of safety issues and resolution are performed. In the last part of this thesis, we propose a new algorithm and try to improve the computational efficiency of the arrival aircraft scheduling problem in the TMA. Then, we summarize the contribution of each part of this thesis.

6.1.1 Arrival air traffic optimization in terminal area under uncertainty

In the first problem, we addressed an arrival aircraft scheduling problem in the deterministic case and the case under uncertainty. A novel probabilistic model is proposed based on the deterministic model. The uncertainty is managed by representing the time information used for conflict detection as random variables. Considering the uncertainty propagation in real operations, the random variables are characterized based on the Markov assumption. By assuming that the prediction error is normally distributed, we benefit from the characteristics of the normal distribution, thus generating an effective method to evaluate the probability of conflict between every two continuously operated aircraft at specific

positions. The objective function is formulated as the expected number of conflicts of all flights in the network and it can be analytically computed. Thanks to the normal distribution and the Markov assumptions, the computational complexity is reduced compared to the typical uncertainty optimization approaches. The time decomposition and sliding window frame combined with simulated annealing is used as solution algorithm. For the proposed probabilistic model, the sliding window frame also plays an important role in limiting the variances of the random variables to a reasonable range. The work is enriched by introducing a Monte-Carlo simulation framework with different scenarios representing the variability of the uncertainty elements. Except for the proposed model, a deterministic model without considering the uncertainty and a model incorporating separation buffers are presented for the purpose of comparison. The solutions of the three models are evaluated by means of stochastic perturbations for 12 windows with different traffic densities. Statistical analyses are conducted by comparing the conflicts generated in different scenarios in the simulation. A comparison is conducted by using the simulation results associated with the deterministic model as the basis upon which the conflict absorption percentages for the other two models are computed. Numerical results show the limitation of the model that incorporates separation buffer. As in some cases, it performs even worse than deterministic model. In contrast, the proposed model provides a certain level of stability in conflict absorption. Moreover, the type of the distributions of the perturbation does not cause notable differences in the performance of conflict absorption. The proposed method has advantages compared to other uncertainty optimization approaches. For example, robust optimization attempts to seek an optimal solution under the worst-case scenario, which leads to a conservative result. This conservatism will lead to a waste of available time slots in arrival management, thus reducing efficiency. However, the proposed approach does not apply hard constraints, so that the balance between the available resources and current traffic density can be achieved while considering robustness for the final result. Though belonging to the stochastic problem, in the proposed approach, the assumption of normal distribution of the random variables eases the computational burden of using scenario-based solution approach (e.g. Sample Average Approach (SAA)). Therefore, it can have a broad application.

6.1.2 Arrival air traffic optimization in an extended horizon

In the second problem, the continuity of the aircraft operation in enroute phase and approach phase enables spreading effects of the early decisions. The extended management horizon is then applied to conduct an early sequencing and trajectory adjustment before aircraft get into the TMA. The common enroute instructions delivered from the controller to the pilots are considered as the decision variables, which are speed adjustment and route assignment. Decision variables for the operations in the TMA are the runway and the TMA entry speed. Due to the large investigated horizon, accurate predictions should be made to make sure the decision-making process is reliable in a complex environment. A dynamic control model applying the technique of time-based rolling horizon is implemented. The horizon is defined as a short time period to indicate the information update. In this model, the information update includes the aircraft that get in and out of the system, and aircraft position update. Through periodical information updates, the decision making

tool accurately captures the current situation and makes predictions to provide associated advisories for controllers. In order to achieve overall situation awareness, for each aircraft, the trajectory predictions are conducted from their current position until landing. Enroute congestion, merging conflict resolution workload are evaluated in the enroute segment, and conflicts are measured in the TMA. The weighted sum of these metrics represents the objective function. Moreover, a dynamic weighted approach is introduced to emphasize the fact that as aircraft get close to the TMA, its situation in TMA becomes more and more important. For each aircraft, this approach allocates weights to the issues that belong to the enroute segment and TMA respectively depending on the aircraft locations with respect to the whole route length. Based on this framework, the algorithm is established by solving the sub-problems associated with each information update. The Simulated Annealing (SA) is implemented and we take the arrival traffic of the Paris Charles de Gaulle (CDG) airport for the case study. Results are evaluated in terms of the performance of safety issues mitigation and the time transferring on the enroute phase to reduce holding at the TMA. We can observe that with a small transfer time of each aircraft, the conflicts in the TMA can be almost eliminated, which suggests the effectiveness and application perspective of the method.

6.1.3 An improved algorithm for arrival aircraft scheduling problem in the terminal area

In the last part of this thesis, we focused on the algorithmic efficiency improvement, which is of great importance for the decision support tool to adapt to frequent changes in the aircraft operational environment. We consider an aircraft scheduling problem in TMA as case study to evaluate the performance of the proposed algorithm. Similar to the first problem, the TMA entry time and the TMA entry speed are set as decision variables. In this study, apart from the conflicts on the resources, delay and speed deviations are considered with a minor weight in the objective function as well. Based on the hypothesis that aircraft are self organized in the system, the original model is reconsidered. Then, a new model focusing on the single aircraft performance is proposed. Opposite to the original model which sums up the conflicts encountered by all flights in the objective function, the proposed algorithm tries to minimize the maximum cost of a single aircraft. Through this way, underlying communication between irrelevant flights can be reduced. Both models are solved by the SA, considering the differences in the two models, details of the optimization procedures are described. The full-day data of the Paris CDG on one landing runway has been applied as a case study. The results of the two models are presented, where the execution times and the qualities of the results are demonstrated. Results show that the newly proposed algorithm has an absolute advantage over the original one in computational time with a slightly dropped performance in terms of the minimization of the delay and speed deviation.

6.2 Perspectives

Several research directions can be followed in the future.

6.2.1 Statistical uncertainty analysis

In the first problem, several directions can be extended. First, the statistical analysis of real delays of aircraft will be beneficial to the determination of time deviation distribution, therefore providing a more reliable reference for the uncertainty propagation model.

Second, from the perspective of uncertainty optimization, the distributionally robust optimization emphasizes the ambiguity of the probability distributions of the uncertain parameters. Analogous to robust optimization that considers the worst-case parameter realization in the uncertainty set, distributionally robust optimization seeks to protect against the worst-case with ambiguous probability distribution. In this case, the probability distribution for the uncertain parameter is unknown but belongs to an infinite set of predefined probability distributions (Rahimian and Mehrotra, 2019). Distributionally robust optimization therefore offers a very interesting alternative to deal with the problem under uncertainty and indicates potential direction for future study.

6.2.2 Multi-objective optimization problem

The design of the algorithms in the current studies is single objective. However, the Air Traffic Management (ATM) involves many stakeholders with different interests, which yields multi-objective optimization problems. Based on the current study context, interest factors such as delay, fuel consumption, equities etc. can be incorporated into a multi-objective optimization model. A direct way to tackle this multi-objective optimization problem, by using our approach, is to consider the weighted sum of all the factors, as the preference or importance of the concerned factors can be reflected on the associated weights. But a more interesting direction is to search the Pareto frontier, in which the compromised relations among the interest factors can be obtained.

6.2.3 Scenario extension and further application of the single performance based algorithm

For the single aircraft performance based algorithm, the future work lies in providing more tests for different scenarios and conducting a deep investigation on the attributes of this algorithm. We hope that it can be applied to the online pre-tactical support tool. Moreover, it may be very useful in the Unmanned Traffic Management (UTM) framework and can be applied to the problems in this field.

6.3 Publications

- **Ying Huo**, Daniel Delahaye, and Mohammed Sbihi,"A Probabilistic Model Based Optimization for Aircraft Scheduling in Terminal Area under Uncertainty," **Transportation Research Part C: Emerging Technologies 132(2021): 103374**.
- **Ying Huo**, Daniel Delahaye, and Mohammed Sbihi,"A Dynamic Control Method for Extended Arrival Management," **Submitted to Transportation Research Part C: Emerging Technologies**.

- Aniket Deshmukh, **Ying Huo**, Daniel Delahaye, hilippe Notry, Mohammed Sbihi, "Algorithmic Efficiency Comparison of Centralized and Distributed Arrival Management (AMAN) Problem in Terminal Airspace," **SID 2020**, 10th SESAR Innovation DaysDec 2020, Virtual event, France.

- **Ying Huo**, Daniel Delahaye, and Mohammed Sbihi," Air Traffic Flow Management under Uncertainty in Terminal Maneuvering Area," **ICRAT 2020**, 9th International Conference for Research in Air Transportation, Jun 2020, Tampa, United States.

- **Ying Huo**, Daniel Delahaye, Ji MA, and Mohammed Sbihi, "Integrated Traffic Flow Based Optimization of Airport and Terminal Area," **SID 2019**, 9th SESAR Innovation Days, Dec 2019, Athenes, Greece

Bibliography

Amrahov, S. E. and Alsalihe, T. A. I. (2011), 'Greedy algorithm for the scheduling aircraft landings', In *5th International Conference on Application of Information and Communication Technologies (AICT)*, IEEE, pp. 1–3.

Balakrishnan, H. and Chandran, B. (2006), 'Scheduling aircraft landings under constrained position shifting', In *AIAA guidance, Navigation, and Control Conference and Exhibit*, p. 6320.

Balakrishnan, H. and Chandran, B. G. (2010), 'Algorithms for scheduling runway operations under constrained position shifting', *Operations Research* **58**(6), 1650–1665.

Beasley, J. E., Krishnamoorthy, M., Sharaiha, Y. M. and Abramson, D. (2000), 'Scheduling aircraft landings—The static case', *Transportation Science* **34**(2), 180–197.

Beasley, J. E., Krishnamoorthy, M., Sharaiha, Y. M. and Abramson, D. (2004), 'Displacement problem and dynamically scheduling aircraft landings', *Journal of the Operational Research Society* **55**(1), 54–64.

Beasley, J. E., Sonander, J. and Havelock, P. (2001), 'Scheduling aircraft landings at london heathrow using a population heuristic', *Journal of the Operational Research Society* **52**(5), 483–493.

Ben-Tal, A. and Nemirovski, A. (2002), 'Robust optimization–methodology and applications', *Mathematical Programming* **92**(3), 453–480.

Bencheikh, G., Boukachour, J. and Alaoui, A. E. H. (2011), 'Improved ant colony algorithm to solve the aircraft landing problem', *International Journal of Computer Theory and Engineering* **3**(2), 224.

Bencheikh, G., Boukachour, J. and Alaoui, A. E. H. (2016), 'A memetic algorithm to solve the dynamic multiple runway aircraft landing problem', *Journal of King Saud University-Computer and Information Sciences* **28**(1), 98–109.

Bencheikh, G., El Khoukhi, F., Baccouche, M., Boudebous, D., Belkadi, A. and Ouahman, A. A. (2013), 'Hybrid algorithms for the multiple runway aircraft landing problem.', *IJCSA* **10**(2), 53–71.

Bennell, J. A., Mesgarpour, M. and Potts, C. N. (2011), 'Airport runway scheduling', *4OR* **9**(2), 115–138.

Bennell, J. A., Mesgarpour, M. and Potts, C. N. (2017), 'Dynamic scheduling of aircraft landings', *European Journal of Operational Research* **258**(1), 315–327.

Bertsimas, D. and Sim, M. (2003), 'Robust discrete optimization and network flows', *Mathematical Programming* **98**(1-3), 49–71.

Bianco, L., Dell'Olmo, P. and Giordani, S. (2006), 'Scheduling models for air traffic control in terminal areas', *Journal of Scheduling* **9**(3), 223–253.

Bianco, L., Nicoletti, B. and Ricciardelli, S. (1978), *An algorithm for optimal sequencing of aircraft in the near terminal area*, Springer, pp. 443–453.

Bosson, C., Xue, M. and Zelinski, S. (2014), 'Optimizing integrated terminal airspace operations under uncertainty', In *2014 IEEE/AIAA 33rd Digital Avionics Systems Conference (DASC)*, IEEE, pp. 1A3–1.

Boursier, L., Favennec, B., Hoffman, E., Trzmiel, A., Vergne, F. and Zeghal, K. (2007), 'Merging arrival flows without heading instructions', In *7th USA/Europe Air Traffic Management R&D Seminar*, pp. 1–8.

Breitsamter, C. (2011), 'Wake vortex characteristics of transport aircraft', *Progress in Aerospace Sciences* **47**(2), 89–134.

Briskorn, D. and Stolletz, R. (2014), 'Aircraft landing problems with aircraft classes', *Journal of Scheduling* **17**(1), 31–45.

Câmara, Á., Rúbio, T. R., Silva, D. C. and Oliveira, E. (2016), 'A comparative study of meta-heuristics for the aircraft landing scheduling problem', In *2016 11th Iberian Conference on Information Systems and Technologies (CISTI)*, IEEE, pp. 1–4.

Caron, G. M., Savéant, P. and Schoenauer, M. (2013), 'Multiobjective tactical planning under uncertainty for air traffic flow and capacity management', In *2013 IEEE Congress on Evolutionary Computation*, IEEE, pp. 1548–1555.

Chandran, B. and Balakrishnan, H. (2007), 'A dynamic programming algorithm for robust runway scheduling', In *2007 American Control Conference*, IEEE, pp. 1161–1166.

Charnes, A. and Cooper, W. W. (1959), 'Chance-constrained programming', *Management Science* **6**(1), 73–79.

Chen, J., Chen, L. and Sun, D. (2017), 'Air traffic flow management under uncertainty using chance-constrained optimization', *Transportation Research Part B: Methodological* **102**, 124–141.

Ciesielski, V. and Scerri, P. (1998), 'Real time genetic scheduling of aircraft landing times', In *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No. 98TH8360)*, IEEE, pp. 360–364.

Clarke, J.-P. B., Ho, N. T., Ren, L., Brown, J. A., Elmer, K. R., Tong, K.-O. and Wat, J. K. (2004), 'Continuous descent approach: Design and flight test for Louisville International Airport', *Journal of Aircraft* **41**(5), 1054–1066.

Cottrell, W. D. (1999), 'Simplified program evaluation and review technique (PERT)', *Journal of Construction Engineering and Management* **125**(1), 16–22.

D'Ariano, A., D'Urgolo, P., Pacciarelli, D. and Pranzo, M. (2010), 'Optimal sequencing of aircrafts take-off and landing at a busy airport', In *13th International IEEE Conference on Intelligent Transportation Systems*, IEEE, pp. 1569–1574.

D'Ariano, A., Pacciarelli, D., Pistelli, M. and Pranzo, M. (2015), 'Real-time scheduling of aircraft arrivals and departures in a terminal maneuvering area', *Networks* **65**(3), 212–227.

D'Ariano, A., Pistelli, M. and Pacciarelli, D. (2012), 'Aircraft retiming and rerouting in vicinity of airports', *IET Intelligent Transport Systems* **6**(4), 433–443.

Dear, R. G. (1976), The dynamic scheduling of aircraft in the near terminal area, Technical report, Flight Transportation Laboratory, Massachusetts Institute.

Delahaye, D., Chaimatanan, S. and Mongeau, M. (2019), *Simulated annealing: From basics to applications*, Springer.

Delahaye, D. and Puechmorel, S. (2013), *Modeling and optimization of air traffic*, Hoboken, NJ: Wiley-ISTE.

Delgado, L. and Prats, X. (2009), 'Fuel consumption assessment for speed variation concepts during the cruise phase', In *ATM Economics Conference*.

Delgado, L. and Prats, X. (2012), 'En route speed reduction concept for absorbing air traffic flow management delays', *Journal of Aircraft* **49**(1), 214–224.

Delgado, L., Prats, X. and Sridhar, B. (2013), 'Cruise speed reduction for ground delay programs: A case study for San Francisco International Airport arrivals', *Transportation Research Part C: Emerging Technologies* **36**, 83–96.

Diallo, C., Ndiaye, B. M. and Seck, D. (2012), 'Scheduling aircraft landings at LSS airport', *Operational Research* **2**, 235–241.

Ernst, A. T., Krishnamoorthy, M. and Storer, R. H. (1999), 'Heuristic and exact algorithms for scheduling aircraft landings', *Networks: An International Journal* **34**(3), 229–241.

Esche, E., You, B. and Repke, J.-U. (2019), *Optimal Design Via Chance-Constrained or Two-Stage Stochastic Programming*, Vol. 47, Elsevier, pp. 169–174.

Eun, Y., Hwang, I. and Bang, H. (2010), 'Optimal arrival flight sequencing and scheduling using discrete airborne delays', *IEEE Transactions on Intelligent Transportation Systems* **11**(2), 359–373.

EUROCONTROL (2013), 'European PBN route spacing handbook', `https://www.eurocontrol.int/sites/default/files/2020-04/eurocontrol-european-pbn-route-spacing-handbook-3.pdf`. Accessed: June-2013.

EUROCONTROL (2019), 'Point Merge - Improving and harmonising arrival operations', `https://www.eurocontrol.int/concept/point-merge`.

EUROCONTROL (2020*a*), 'EUROCONTROL five-year forecast 2020-2024', `https://www.eurocontrol.int/publication/eurocontrol-five-year-forecast-2020-2024`. Accessed: Nov-2020.

EUROCONTROL (2020*b*), 'EUROCONTROL revises downwards draft traffic scenarios for September to February 2021', "`https://www.eurocontrol.int/sites/default/files/2020-09/eurocontrol-draft-traffic-scenarios-14092020.pdf`". Accessed: 14-Sep-2020.

Farhadi, F., Ghoniem, A. and Al-Salem, M. (2014), 'Runway capacity management–an empirical study with application to Doha International Airport', *Transportation Research Part E: Logistics and Transportation Review* **68**, 53–63.

Faye, A. (2018), 'A quadratic time algorithm for computing the optimal landing times of a fixed sequence of planes', *European Journal of Operational Research* **270**(3), 1148–1157.

Frankovich, M. J. *Air traffic flow management at airports: A unified optimization approach*. PhD thesis, Massachusetts Institute of Technology.

Furini, F., Kidd, M. P., Persiani, C. A. and Toth, P. (2014), 'State space reduced dynamic programming for the aircraft sequencing problem with constrained position shifting', In *International Symposium on Combinatorial Optimization*, Springer, pp. 267–279.

Furini, F., Kidd, M. P., Persiani, C. A. and Toth, P. (2015), 'Improved rolling horizon approaches to the aircraft sequencing problem', *Journal of Scheduling* **18**(5), 435–447.

Gilbo, E. P., Smith, S. B. *et al.* (2011), 'New method for probabilistic traffic demand predictions for en route sectors based on uncertain predictions of individual flight events', In *9th USA/Europe Air Traffic Management Research and Development Seminar (ATM2011), Berlin*.

Girish, B. (2016), 'An efficient hybrid particle swarm optimization algorithm in a rolling horizon framework for the aircraft landing problem', *Applied Soft Computing* **44**, 200–221.

Gonze, F., Huens, E., Jungers, R. M., Simonetto, A. and Boucquey, J. (2018), 'Probabilistic occupancy counts and flight criticality measures in air traffic management', *Journal of Air Transportation* **26**(3), 94–103.

Gudmundsson, S. V., Cattaneo, M. and Redondi, R. (2021), 'Forecasting temporal world recovery in air transport markets in the presence of large economic shocks: The case of covid-19', *Journal of Air Transport Management* **91**, 102007.

Heidt, A., Helmke, H., Kapolke, M., Liers, F. and Martin, A. (2016), 'Robust runway scheduling under uncertain conditions', *Journal of Air Transport Management* **56**, 28–37.

Heidt, A., Helmke, H., Liers, F. and Martin, A. (2014), 'Robust runway scheduling using a time-indexed model', In *Fourth SESAR Innovation Days*, pp. 1–8.

Hu, X.-B. and Chen, W.-H. (2005), 'Genetic algorithm based on receding horizon control for arrival sequencing and scheduling', *Engineering Applications of Artificial Intelligence* **18**(5), 633–642.

Hu, X. and Di Paolo, E. (2008), 'Binary-representation-based genetic algorithm for aircraft arrival sequencing and scheduling', *IEEE Transactions on Intelligent Transportation Systems* **9**(2), 301–310.

Hu, X. and Di Paolo, E. (2009), 'An efficient genetic algorithm with uniform crossover for air traffic control', *Computers & Operations Research* **36**(1), 245–259.

Huang, H. and Tomlin, C. (2009), 'A network-based approach to en-route sector aircraft trajectory planning', In *AIAA Guidance, Navigation, and Control Conference*, p. 6169.

Huo, Y., Delahaye, D. and Sbihi, M. (2021), 'A probabilistic model based optimization for aircraft scheduling in terminal area under uncertainty', *Transportation Research Part C: Emerging Technologies* **132**, 103374.

ICAO (2008), Performance-based navigation (PBN) manual (Doc9613), Technical report.

ICAO (2016), Air traffic management, Doc4444, Technical report.

ICAO (2017), 'Air traffic services', `http://skyrise.aero/wp-content/uploads/2017/03/ICAO-Annex-11-Air-traffic-services.pdf`. Accessed: March-2017.

ICAO (2021*a*), 'Effects of novel coronavirus (COVID-19) on civil aviation: economic impact analysis', `https://www.icao.int/sustainability/Documents/Covid-19/ICAO_coronavirus_Econ_Impact.pdf`. Accessed: Jan-2021.

ICAO (2021*b*), 'Forecasts of scheduled passenger and freight traffic', `https://www.icao.int/sustainability/Pages/eap-fp-forecast-scheduled-passenger-traffic.aspx`. Accessed: 29-June-2021.

Ikli, S., Mancel, C., Mongeau, M., Olive, X. and Rachelson, E. (2021), 'The aircraft runway scheduling problem: A survey', *Computers & Operations Research* p. 105336.

John, R. and Kamgarpour, M. (2010), 'Benefits of continuous descent operations in high-density terminal airspace considering scheduling constraints', In *10th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, p. 9115.

Jones, J. C., Lovell, D. J. and Ball, M. O. (2013), 'En route speed control methods for transferring terminal delay', In *10th USA/Europe Air Traffic Management Research and Development Seminar (ATM2013), Chicago*.

Kapolke, M., Fürstenau, N., Heidt, A., Liers, F., Mittendorf, M. and Weiß, C. (2016), 'Pre-tactical optimization of runway utilization under uncertainty', *Journal of Air Transport Management* **56**, 48–56.

Khassiba, A., Bastin, F., Cafieri, S., Gendron, B. and Mongeau, M. (2020), 'Two-stage stochastic mixed-integer programming with chance constraints for extended aircraft arrival management', *Transportation Science* **54**(4), 897–919.

Khassiba, A., Bastin, F., Gendron, B., Cafieri, S. and Mongeau, M. (2019), 'Extended aircraft arrival management under uncertainty: A computational study', *Journal of Air Transportation* **27**(3), 131–143.

Kirkpatrick, S., Gelatt, C. D. and Vecchi, M. P. (1983), 'Optimization by simulated annealing', *Science* **220**(4598), 671–680.

Kleywegt, A. J., Shapiro, A. and Homem-de Mello, T. (2002), 'The sample average approximation method for stochastic discrete optimization', *SIAM Journal on Optimization* **12**(2), 479–502.

Kupfer, M. (2009), 'Scheduling aircraft landings to closely spaced parallel runways', In *8th USA/Europe Air Traffic Management Research and Development Seminar (ATM 2009), Napa, CA*.

Lee, H. and Balakrishnan, H. (2008), 'A study of tradeoffs in scheduling terminal-area operations', *Proceedings of the IEEE* **96**(12), 2081–2095.

Li, Z. and Ierapetritou, M. (2008), 'Process scheduling under uncertainty: Review and challenges', *Computers & Chemical Engineering* **32**(4-5), 715–727.

Liang, M., Delahaye, D. and Marechal, P. (2018), 'Conflict-free arrival and departure trajectory planning for parallel runway with advanced point-merge system', *Transportation Research Part C: Emerging Technologies* **95**, 207–227.

Lieder, A., Briskorn, D. and Stolletz, R. (2015), 'A dynamic programming approach for the aircraft landing problem with aircraft classes', *European Journal of Operational Research* **243**(1), 61–69.

Lieder, A. and Stolletz, R. (2016), 'Scheduling aircraft take-offs and landings on interdependent and heterogeneous runways', *Transportation Research Part E: Logistics and Transportation Review* **88**, 167–188.

Ma, J., Delahaye, D., Sbihi, M. and Mongeau, M. (2016), 'Merging flows in terminal maneuvering area using time decomposition approach', In *7th International Conference on Research in Air Transportation (ICRAT 2016)*.

Ma, J., Delahaye, D., Sbihi, M., Scala, P. and Mota, M. A. M. (2019), 'Integrated optimization of terminal maneuvering area and airport at the macroscopic level', *Transportation Research Part C: Emerging Technologies* **98**, 338–357.

Marceau Caron, G. *Optimization and uncertainty handling in air traffic management.* PhD thesis, Université Paris 11.

Marla, L., Rikun, A., Stauffer, G. and Pratsini, E. (2020), 'Robust modeling and planning: Insights from three industrial applications', *Operations Research Perspectives* **7**, 100150.

Mercer, J., Bienert, N., Gomez, A., Hunt, S., Kraut, J. M., Martin, L., Morey, S., Green, S., Prevôt, T. and Wu, M. G. (2013), 'The impact of trajectory prediction uncertainty on air traffic controller performance and acceptability', In *2013 Aviation Technology, Integration, and Operations Conference*, p. 4320.

Meric, O. S. and Turan, O. (2016), 'Evaluation of aircraft descent profile', *Energy Procedia* **95**, 308–313.

Messaoud, M. B. (2020), 'A thorough review of aircraft landing operation from practical and theoretical standpoints at an airport which may include a single or multiple runways', *Applied Soft Computing* p. 106853.

Messaoud, M. B., Ghedira, K. and Kefi, M. (2018), 'Detailed mathematical programming formulations for the aircraft landing problem on a single and multiple runway configurations', *Procedia Computer Science* **126**, 345–354.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. and Teller, E. (1953), 'Equation of state calculations by fast computing machines', *The Journal of Chemical Physics* **21**(6), 1087–1092.

Meyn, L. and Erzberger, H. (2005), 'Airport arrival capacity benefits due to improved scheduling accuracy', In *AIAA 5th ATIO and16th Lighter-Than-Air Sys Tech. and Balloon Systems Conferences*, p. 7376.

Mokhtarimousavi, S., Rahami, H. and Kaveh, A. (2015), 'Multi-objective mathematical modeling of aircraft landing problem on a runway in static mode, scheduling and sequence determination using nsga-ii', *Iran University of Science & Technology* **5**(1), 21–36.

Murça, M. C. R. and Müller, C. (2015), 'Control-based optimization approach for aircraft scheduling in a terminal area with alternative arrival routes', *Transportation Research Part E: Logistics and Transportation Review* **73**, 96–113.

Ng, K., Lee, C., Chan, F. T. and Qin, Y. (2017), 'Robust aircraft sequencing and scheduling problem with arrival/departure delay using the min-max regret approach', *Transportation Research Part E: Logistics and Transportation Review* **106**, 115–136.

Nikoleris, A., Chatterji, G., Almog, N. and Palopo, K. (2012), 'Arrival delay absorption using extended metering with speed control', In *12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference and 14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, p. 5640.

Nikoleris, A. and Erzberger, H. (2014), 'Autonomous system for air traffic control in terminal airspace', In *14th AIAA Aviation Technology, Integration, and Operations Conference*, p. 2861.

Peprah, A. K., Appiah, S. K., Amponsah, S. K. *et al.* (2017), 'An optimal cooling schedule using a simulated annealing based approach', *Applied Mathematics* **8**(08), 1195.

Peters, A. (2017), 'Applicability of extended arrival manager to assist ANSPs in determining an arrival sequence that accounts for en-route delays', *Preprint submitted to Lecturers Aviation* .

Pinol, H. and Beasley, J. (2006), 'Scatter search and bionomic algorithms for the aircraft landing problem', *European Journal of Operational Research* **171**(2), 439–462.

Prakash, R., Piplani, R. and Desai, J. (2018), 'An optimal data-splitting algorithm for aircraft scheduling on a single runway to maximize throughput', *Transportation Research Part C: Emerging Technologies* **95**, 570–581.

Psaraftis, H. N. (1978), A dynamic programming approach to the aircraft sequencing problem, Technical report, Cambridge, Massachusetts Institute of Technology.

Rahimian, H. and Mehrotra, S. (2019), 'Distributionally robust optimization: A review', *arXiv preprint arXiv:1908.05659* .

Sahinidis, N. V. (2004), 'Optimization under uncertainty: state-of-the-art and opportunities', *Computers & Chemical Engineering* **28**(6-7), 971–983.

Salehipour, A. (2020), 'An algorithm for single-and multiple-runway aircraft landing problem', *Mathematics and Computers in Simulation* **175**, 179–191.

Samà, M., D'Ariano, A., Corman, F. and Pacciarelli, D. (2017), 'Metaheuristics for efficient aircraft scheduling and re-routing at busy terminal control areas', *Transportation Research Part C: Emerging Technologies* **80**, 485–511.

Samà, M., D'Ariano, A., D'Ariano, P. and Pacciarelli, D. (2014), 'Optimal aircraft scheduling and routing at a terminal control area during disturbances', *Transportation Research Part C: Emerging Technologies* **47**, 61–85.

Samà, M., D'Ariano, A., D'Ariano, P. and Pacciarelli, D. (2017), 'Scheduling models for optimal aircraft traffic control at busy airports: tardiness, priorities, equity and violations considerations', *Omega* **67**, 81–98.

Samà, M., D'Ariano, A. and Pacciarelli, D. (2013), 'Rolling horizon approach for aircraft scheduling in the terminal control area of busy airports', *Procedia-Social and Behavioral Sciences* **80**, 531–552.

Scala, P., Mota, M. M., Ma, J. and Delahaye, D. (2019), 'Tackling uncertainty for the development of efficient decision support system in air traffic management', *IEEE Transactions on Intelligent Transportation Systems* **21**(8), 3233–3246.

Scala, P., Mota, M. M., Wu, C.-L. and Delahaye, D. (2021), 'An optimization–simulation closed-loop feedback framework for modeling the airport capacity management problem under uncertainty', *Transportation Research Part C: Emerging Technologies* **124**, 102937.

SESAR (2015*a*), 'Controlled time of arrival (CTA) in medium density and complexity environments', `https://www.sesarju.eu/node/2199`.

SESAR (2015*b*), 'Extended arrival manager (E-AMAN)', `https://op.europa.eu/en/publication-detail/-/publication/fe9eb858-60b4-496c-82db-6854d05a699e`. Accessed: 27-April-215.

SESAR (2019*a*), 'From innovation to deployment'. `https://www.ecologie.gouv.fr/sites/default/files/sesar_dsna_world_atm_congress_2019.pdf`, Accessed 2019.

SESAR (2019*b*), Sesar solution catalogue 2019, Technical report.

SESAR-FAA (2017), NextGen - SESAR state of harmonisation, Technical report, Publications office of the Europe Union.

Soomer, M. J. and Franx, G. J. (2008), 'Scheduling aircraft landings using airlines' preferences', *European Journal of Operational Research* **190**(1), 277–291.

Swenson, H. N., Thipphavong, J., Sadovsky, A., Chen, L., Sullivan, C. and Martin, L. (2011), 'Design and evaluation of the terminal area precision scheduling and spacing system', In *USA/Europe Air Traffic Management Research and Development Seminar (ATM2011)*, number ARC-E-DAA-TN2863.

Tang, K., Wang, Z., Cao, X. and Zhang, J. (2008), 'A multi-objective evolutionary approach to aircraft landing scheduling problems', In *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*, IEEE, pp. 3650–3656.

Tielrooij, M., Borst, C., Van Paassen, M. M. and Mulder, M. (2015), 'Predicting arrival time uncertainty from actual flight information', In *Proceedings of the 11th USA/Europe Air Traffic Management Research and Development Seminar*, pp. 577–586.

Usanmaz, O., Sınar, G. and Sahin, O. (2019), 'An assessment of a proposed arrival route model in terminal airspace', *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* **233**(11), 4032–4038.

Vadlamani, S. and Hosseini, S. (2014), 'A novel heuristic approach for solving aircraft landing problem with single runway', *Journal of Air Transport Management* **40**, 144–148.

Wanke, C., Mulgund, S., Greenbaum, D. and Song, L. (2004), 'Modeling traffic prediction uncertainty for traffic management decision support', In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, p. 5230.

Xu, B. (2017), 'An efficient ant colony algorithm based on wake-vortex modeling method for aircraft scheduling problem', *Journal of Computational and Applied Mathematics* **317**, 157–170.

Zhan, Z., Zhang, J., Li, Y., Liu, O., Kwok, S., Ip, W. and Kaynak, O. (2010), 'An efficient ant colony system based on receding horizon control for the aircraft arrival sequencing and scheduling problem', *IEEE Transactions on Intelligent Transportation Systems* **11**(2), 399–412.

Zhang, J., Zhao, P., Zhang, Y., Dai, X. and Sui, D. (2020), 'Criteria selection and multi-objective optimization of aircraft landing problem', *Journal of Air Transport Management* **82**, 101734.

Zuniga, C. A., Piera, M. A., Ruiz, S. and Del Pozo, I. (2013), 'A CD&CR causal model based on path shortening/path stretching techniques', *Transportation Research Part C: Emerging Technologies* **33**, 238–256.

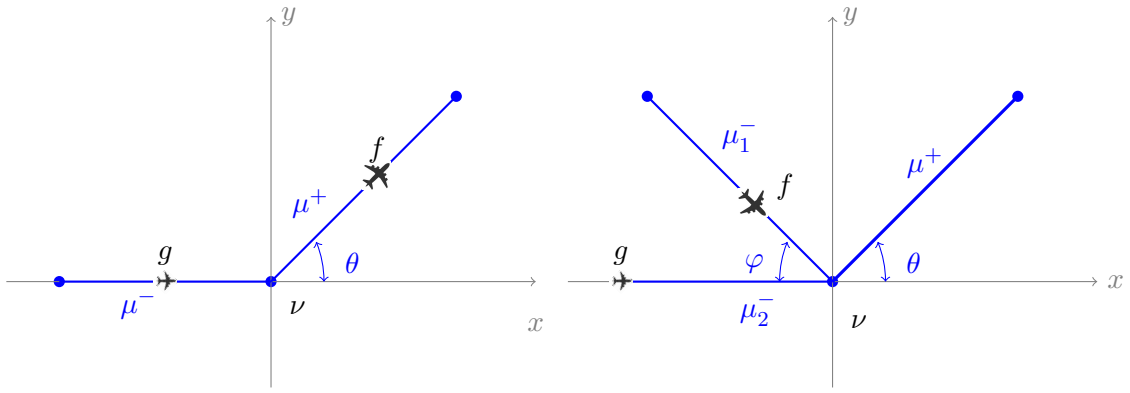
Zuniga, C., Delahaye, D. and Piera, M. A. (2011), 'Integrating and sequencing flows in terminal maneuvering area by evolutionary algorithms', In *2011 IEEE/AIAA 30th Digital Avionics Systems Conference*, IEEE, pp. 2A1–1.

Appendix A

Proof of Lemmas

In this appendix we prove Lemma 1 and Lemma 2 for the conflict evaluation in Chapter 3 and Chapter 4. The following assumption is proposed for our study case.

Assumption 1 *For any two distinct adjacent links $\mu_1 = (\nu_1, \nu)$ and $\mu_2 = (\nu_2, \nu)$ (or $\mu_2 = (\nu, \nu_2)$), the distance from ν_2 to the link μ_1 and from ν_1 to μ_2 are greater than or equal to h (horizontal separation requirement). By the distance from a point to a link, we mean the length of the shortest line joining the point to a point on the link.*



(a) Aircraft f and g arrive at node ν through the same link.

(b) Aircraft f and g converge to node ν from different links.

Figure A.1: Node conflict detection scenarios regarding different trajectory interaction patterns of the flight pair.

A.1 Proof of Lemma 1

To shorten the notation let us denote the speed of aircraft f by v_1 instead of $v_{f,\nu}$ and the speed of aircraft g by v_2 instead of $v_{g,\nu}$. We assume that aircraft g passes the node ν after δ units of time when f passed it, that is $\delta = t_{g,\nu} - t_{f,\nu}$. Consider first the case where both aircraft enter node ν through the same link μ^- of length l^- then fly along the link μ^+ of

length l^+ (see Fig. A.1a). We are interested in the occurrence of a node conflict at ν *i.e.* when f flies on μ^+ and g flies on μ^-. If $\delta > \frac{l^-}{v_2} + \frac{l^+}{v_1}$, that is to say, if aircraft f leaves the link μ^+ before aircraft g enters the link μ^-, then the aircraft are not in conflict. Hence we can conclude a prerequisite condition, which is $\delta \leq \frac{l^-}{v_2} + \frac{l^+}{v_1}$. Under this condition, the (square of) distance between the two aircraft after t units of time when aircraft f passed the node ν is given by:

$$d^2(t,\delta) = (t(v_1\cos\theta - v_2) + \delta v_2)^2 + t^2 v_1^2 \sin^2\theta, \tag{A.1}$$

which has to be considered only during the presence of the two aircraft on their respective links, that is $t \in [\max(0, \delta - \frac{l^-}{v_2}), \min(\frac{l^+}{v_1}, \delta)]$. Let us define the function m representing the minimum (squared) distance between the two aircraft during this period, or more precisely

$$m(\delta) := \min_{t\in[\max(0,\delta-\frac{l^-}{v_2}),\min(\frac{l^+}{v_1},\delta)]} d^2(t,\delta). \tag{A.2}$$

The time to achieve the minimum (square) distance is computed by obtaining the derivative of $d^2(t,\delta)$ with respect to t, then we have:

$$\begin{aligned}\frac{\partial d^2}{\partial t}(t,\delta) &= 2(v_1\cos\theta - v_2)\left(t(v_1\cos\theta - v_2) + \delta v_2\right) + 2tv_1^2\sin^2\theta \\ &= 2\delta v_2(v_1\cos\theta - v_2) + 2t(v_1^2 + v_2^2 - 2v_1v_2\cos\theta),\end{aligned} \tag{A.3}$$

so the second order polynomial with a positive leading coefficient $d^2(\delta,\cdot)$ achieves its minimum on $\mathbb{R}$ at $t^*(\delta)$. By ensuring Eq. A.3 equals 0, we obtain a value $\alpha = \frac{v_2(v_2 - v_1\cos\theta)}{v_1^2+v_2^2-2v_1v_2\cos\theta}$ such that $t^*(\delta) = \alpha\delta$. The following three exclusive cases will be considered successively: $\alpha \leq 0$, $\alpha \geq 1$ and $0 < \alpha < 1$. These cases respectively correspond to the following cases in the definition of the expression of a (Eq. 3.2 in Chapter 3) in Lemma 1:

(i) **Case 1:** $\alpha \leq 0$ which is equivalent to $v_2 \leq v_1\cos\theta$
In this case, we have $t^*(\delta) = \alpha\delta < 0$. The relation between the $d^2(\delta)$ with respect to the time t is shown in Fig. A.2. The minimum distance between the two aircraft $m(\delta)$ is then achieved at $\max(0, \delta - \frac{l^-}{v_2})$ with a value

$$m^{\frac{1}{2}}(\delta) = \begin{cases} \delta v_2 & \text{if } \delta \leq \frac{l^-}{v_2} \\ \geq h, & \text{otherwise.}\end{cases} \tag{A.4}$$

The last inequality follows from the fact that at $t = \delta - \frac{l^-}{v_2}$, the aircraft g is at the entry node of link μ^- and Assumption 1. Consequently $m^{\frac{1}{2}}(\delta) \geq h$ if and only if $\delta \geq \frac{h}{v_2}$.

(ii) **Case 2:** $\alpha \geq 1$ is equivalent to $v_1 \leq v_2 \cos\theta$:

In this case, $t^*(\delta) \geq \delta$, the valid time period $t \in [\max(0, \delta - \frac{l^-}{v_2}), \min(\frac{l^+}{v_1}, \delta)]$ which is smaller or equal to δ. Therefore, we have $t^*(\delta) \geq \min(\delta, \frac{l^+}{v_1})$. The graph of the function $d^2(\delta)$with respect to the time t is displayed in Fig. A.3. The minimum (Eq. A.2) is achieved at $\min(\delta, \frac{l^+}{v_1})$ with a value

$$m^{\frac{1}{2}}(\delta) = \begin{cases} \delta v_1 & \text{if } \delta \leq \frac{l^+}{v_1} \\ \geq h, & \text{otherwise.} \end{cases} \tag{A.5}$$

The last inequality follows from the fact that at $t = \frac{l^+}{v_1}$ the aircraft f is at the exit node of link μ^+ and Assumption 1. Consequently $m^{\frac{1}{2}}(\delta) \geq h$ if and only if $\delta \geq \frac{h}{v_1}$.

(iii) **Case3:** $0 < \alpha < 1$ is equivalent to $v_1 > v_2 \cos\theta$

In this case, we have

- If $\delta \in [0, \min\left(\frac{l^-}{(1-\alpha)v_2}, \frac{l^+}{\alpha v_1}\right)]$, then $t^*(\delta) = \alpha\delta \in [\max(0, \delta - \frac{l^-}{v_2}), \min(\frac{l^+}{v_1}, \delta)]$ which lies in the valid time range, and consequently we have $m^{\frac{1}{2}}(\delta) = d(\alpha\delta, \delta) = \frac{\delta v_1 v_2 \sin\theta}{\sqrt{v_1^2 + v_2^2 - 2v_1 v_2 \cos\theta}}$. This case is represented in Fig. A.4, in which $t^*(\delta)$ is located in the range of the valid time period.
- If $\delta > \min\left(\frac{l^-}{(1-\alpha)v_2}, \frac{l^+}{\alpha v_1}\right)$ we consider two sub-cases : either the last min equals to $\frac{l^+}{\alpha v_1}$ or to $\frac{l^-}{(1-\alpha)v_2}$. In the first sub-case, $\delta > \frac{l^+}{\alpha v_1}$ implies $\delta > \alpha\delta > \frac{l^+}{v_1}$, so the minimum (A.2) is achieved at $\frac{l^+}{v_1}$ when the aircraft f is at the exit node of link μ^+, and thanks to Assumption 1, $m^{\frac{1}{2}}(\delta) \geq h$. In the other sub-case,

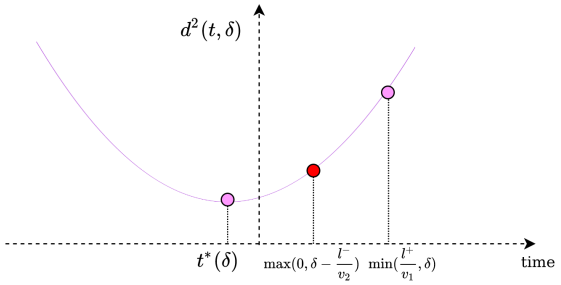


Figure A.2: The distance between the two aircraft with respect to time in case 1.

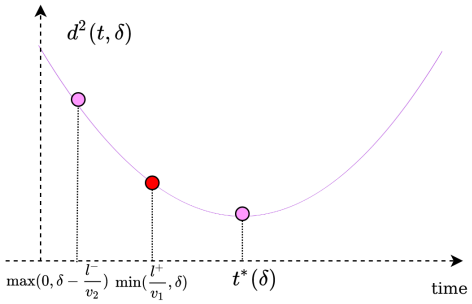


Figure A.3: The distance between the two aircraft with respect to time in case 2.

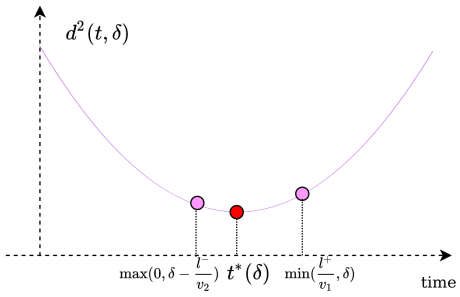


Figure A.4: The distance between the two aircraft with respect to time in case 3.

$\delta > \frac{l^-}{(1-\alpha)v_2}$ implies $\delta - \frac{l^-}{v_2} > \alpha\delta > 0$, so the minimum (Eq. A.2) is achieved at $\delta - \frac{l^-}{v_2}$ when the aircraft g is at the entry node of link μ^-, and again thanks to Assumption 1, $m^{\frac{1}{2}}(\delta) \geq h$.

In sum, we have

$$m^{\frac{1}{2}}(\delta) = \begin{cases} d(\alpha\delta,\delta) = \frac{\delta v_1 v_2 \sin\theta}{\sqrt{v_1^2+v_2^2-2v_1v_2\cos\theta}} & \text{if } \delta \in [0, \min\left(\frac{l^-}{(1-\alpha)v_2}, \frac{l^+}{\alpha v_1}\right)] \\ \geq h, & \text{otherwise.} \end{cases} \tag{A.6}$$

Consequently $m^{\frac{1}{2}}(\delta) \geq h$ if and only if $\delta \geq \frac{h\sqrt{v_1^2+v_2^2-2v_1v_2\cos\theta}}{v_1 v_2 \sin\theta}$.

Gathering all the cases we obtain, $m^{\frac{1}{2}}(\delta) \geq h$, meaning that no horizontal conflict occur between f and g at node ν, if and only if $\delta \geq a$, where a is defined by

$$a = \begin{cases} \frac{h}{v_2} & \text{if } v_2 \leq v_1\cos\theta, \\ \frac{h}{v_1} & \text{if } v_1 \leq v_2\cos\theta, \\ \frac{h\sqrt{v_1^2+v_2^2-2v_1v_2\cos\theta}}{v_1v_2\sin\theta}, & \text{otherwise.} \end{cases} \tag{A.7}$$

Considering now the case where aircraft f and g enter node ν from different links μ_1^- and μ_2^- then fly along a link μ^+ (see Fig. A.1b). A node conflict could happen in one of the two following situations:

- aircraft f flies on μ^+ and aircraft g on link μ_2^-,
- aircraft f flies on μ_1^- and aircraft g on link μ_2^-.

From the above, no conflict can happen in the first situation if and only if $\delta \geq a$, Consider now the second situation. Let us denote by l_i the length of link μ_i^-, $i = 1, 2$. If $\delta > \min(\frac{l_1}{v_1}, \frac{l_2}{v_2})$, that is to say, if aircraft f leaves the link μ_1^- before aircraft g enters the link μ_2^-, then the aircraft are not in conflict. Hence, we assume that $\delta \leq \min(\frac{l_1}{v_1}, \frac{l_2}{v_2})$.

Under this condition, the (square of) distance between the two aircraft t units of time before aircraft f passes node ν is given by:

$$d^2(t,\delta) = (t(v_2 - v_1\cos\varphi) + \delta v_2)^2 + t^2 v_1^2 \sin^2\varphi, \tag{A.8}$$

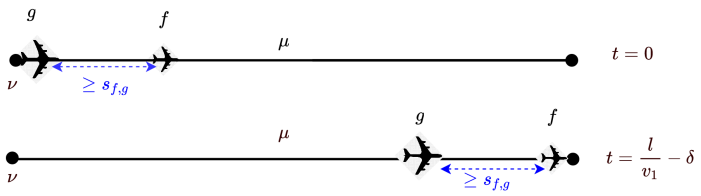


Figure A.5: The required separation time of the two aircraft at the link entry point is computed based on the condition that the wake turbulence separation requirement is satisfied both at the entry and exit point of this link while maintaining the same order.

which has to be considered only during the presence of the two aircraft on their respective links, that is $t \in [0, \min(\frac{l_1}{v_1}, \frac{l_2}{v_2} - \delta)]$. By arguments similar to the first part of the proof, we can establish that $m(\delta) := \min_{t \in [0, \min(\frac{l_1}{v_1}, \frac{l_2}{v_2} - \delta)]} d^2(t, \delta) \geq h^2$ if and only if

$$\delta \geq b := \begin{cases} \frac{h}{v_2} & \text{if } v_1 \cos\varphi \leq v_2, \\ \frac{h\sqrt{v_1^2 + v_2^2 - 2v_1 v_2 \cos\varphi}}{v_1 v_2 \sin\varphi}, & \text{otherwise.} \end{cases} \tag{A.9}$$

In summary, putting the two situations together implies that the absence of conflict is equivalent to $\delta \geq \max(a, b)$.

A.2 Proof of Lemma 2

Note that, in this case, δ denotes the time difference of aircraft f and g passing the link entry point. We remark that if $\delta > \frac{l}{v_1}$, that is to say, if aircraft f leaves link μ before aircraft g enters it, then the aircraft are not in conflict. Hence, assuming that $\delta \leq \frac{l}{v_1}$, the distance between the two aircraft after t units of time that aircraft g has passed node ν is given by:

$$d(t, \delta) = (\delta + t)v_1 - tv_2$$

Then, there is no conflict between f and g at link μ if and only if the wake turbulence separation is satisfied at the entry (at $t = 0$) and the exit (at $t = \frac{l}{v_1} - \delta$) of link μ as shown in the Fig. A.5. From which we can obtain:

$$\delta v_1 \geq s_{f,g} \text{ and } l - v_2(\frac{l}{v_1} - \delta) \geq s_{f,g}$$

or equivalently

$$\delta \geq \max\left(\frac{s_{f,g}}{v_1}, \frac{s_{f,g}}{v_2} + l\frac{v_2 - v_1}{v_1 v_2}\right).$$