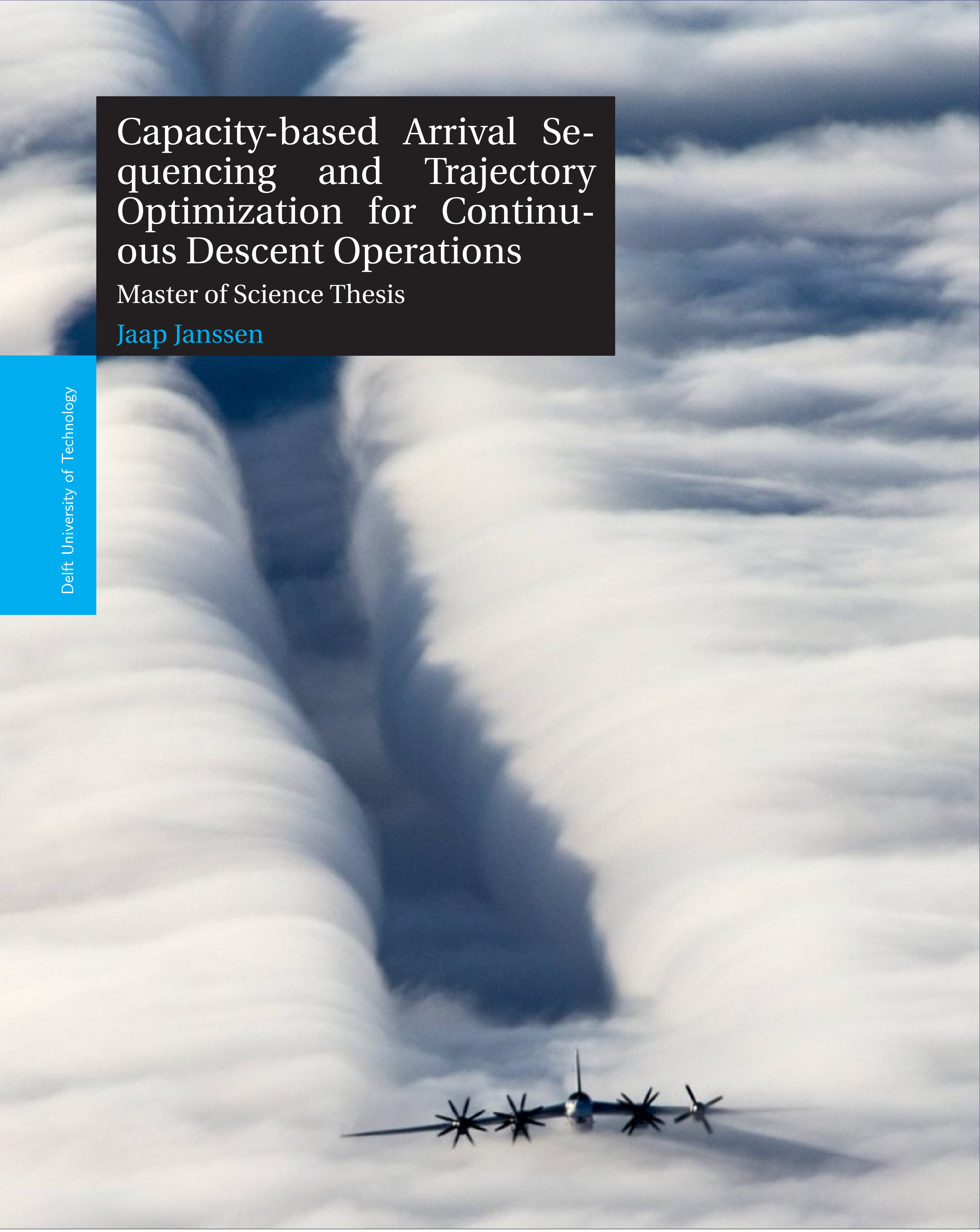
Capacity-based Arrival Sequencing and Trajectory Optimization for Continuous Descent Operations

Master of Science Thesis

Jaap Janssen

Delft University of Technology



Delft University of Technology

Challenge the future

Capacity-based Arrival Sequencing and Trajectory Optimization for Continuous Descent Operations

Master of Science Thesis

by

Jaap Janssen

in partial fulfilment of the requirements for the degree of

Master of Science
in Aerospace Engineering

at the Delft University of Technology,
to be defended publicly on Friday December 6, 2019 at 10:00 AM.

Specialization:	Flight Performance & Propulsion	
Student number:	4111516	
Supervisor:	Ir. P.C. Roling	
Thesis committee:	Prof. dr. R. Curran,	TU Delft
	Ir. J.A. Melkert,	TU Delft

An electronic version of this thesis is available at http://repository.tudelft.nl/.

Abstract

Due to the rapid growth of air traffic and the corresponding increase in demanded landing capacity of airports, much research has been done on arrival scheduling and approach trajectory optimization aiming to not only maximize the effective runway throughput of an airports runway system but also significantly reduce fuel consumption. This thesis research considers both, arrival scheduling and approach trajectory optimization, as one combined optimization problem with the goal of balancing fuel-optimality and time-optimality based on the presently demanded runway throughput in a dynamic approach environment. A hybrid optimization tool is developed, solving the arrival scheduling problem by means of a sequential quadratic programming algorithm while solving the corresponding approach trajectories as an optimal control problem using a hp-adaptive Gaussian quadrature collocation method. Both optimization problems are linked through pre-computed aircraft-specific fuel curves, providing fuel information for each achievable landing time to base scheduling decisions on. The arrival optimization tool applies continuous descent approach procedures over a fixed lateral approach path. Further, the freeze horizon concept is applied which together with constrained position shifting confines the optimization problem down to a size that can be solved in real time. Compared to the currently prevailing first-come first-served scheduling method, a 14% increase in effective runway capacity was achieved consistently while a fuel reduction of 2.2-4.8%was achieved depending on the tested average throughput.

Contents

Nomenclature

Symbols

AC/hr	Aircraft per hour
D	Drag
ff	Fuel Flow
FL	Flight Level (FL100 = 10000ft)
ft	Foot
g0	gravitational acceleration
h	altitude
kg	Kilogram
m	Meter
nm	Nautical mile
s	Second
T	Thrust
tt	Travel Time
V	Airspeed
W	Weight
η	Engine Thrust Setting
γ	Flight Path Angle

Abbreviations

ACAS	Airborne collision avoidance system
ACDA	Advanced Continuous Descent Approach
AMAN	Arrival Manager
ASP	Aircraft Scheduling/Sequencing Problem
ATC	Air Traffic Control Air Traffic Controller
ATM	Air Traffic Management Air Traffic Manager
CDA	Continuous Descent Approach
CPS	Constrained Position Shifting
EAS	Equivalent Air Speed
FAF/FAP	Final Approach Fix / Final Approach Point
FC	Fuel Capacity
FCFS	First Come First Serve
FMS	Flight Management System
GPOPS	Gauss Pseudospectral Optimization Software
IAF	Initial Approach Fix
ICAO	International Civil Aviation Organization
KIAS	Indicated Airspeed [knots]
MPS	Maximum Position Shift
MRS	Minimum Radar Separation

MLW	Maximum Landing Weight
MTOW	Maximum Take Off Weight
MZFW	Maximum Zero Fuel Weight
OEW	Operating Empty Weight
PCHIP	Piecewise Cubic Hermite Interpolating Polynomial
PHCAP	Practical Hourly Capacity
P-RNAV	Precision Area Navigation
RA	Resolution Advisory
SESAR (JU)	Single European Sky ATM Research (Joint Undertaking)
SQP	Sequential Quadratic Programming
SSR	Secondary Surveillance Radar
STAR	Standard Arrival Route
STCA	Short-term conflict alert
TAS	True Airspeed
TBO	Trajectory Based Operations
TDDA	Three-Degree Decelerating Approach
TMA/TCA	Terminal Maneuvering Area / Terminal Control Area
TTA	Target Time of Arrival
TTL	Throughput Threshold Level
WPF	Waypoint Finder

List of Figures

List of Tables

1

Introduction

Due to ever-growing air traffic, airport runway systems are becoming a major bottleneck in the current air traffic control operations since its capacity cannot handle the required traffic throughput. At the same time, noise and emissions in the vicinity of airports are becoming more of a concern. Continuous descent operations have been proven to significantly decrease noise and fuel consumption during an approach but are accompanied by larger required wake separation distances between approaching aircraft, which greatly decreases the number of landings a runway can handle per hour [8]. This leads to an important trade-off between noise and emissions on the one hand and efficient use of runway capacity on the other hand.
The solution that is generally advocated by airport operators to handle the increasing required runway throughput is to increase the capacity by making more efficient use of existing runway facilities, rather than building additional runways [9]. In current airport operations, approaching aircraft land in the order in which they entered the Terminal Maneuvering Area (TMA). This scheduling strategy is often referred to as the First-Come First-Served strategy. However since required wake separation distances vary greatly depending on an aircraft's weight class, it is evident that runway capacity can be used more efficiently by finding more optimal landing sequences. This problem is commonly referred to as the Aircraft Scheduling Problem (ASP). By computing corresponding fuel-optimal and conflict-free approach trajectories for each approaching aircraft, meeting the found optimal landing schedule, the effective runway capacity is increased while at the same time the noise emission and fuel consumption of approaching aircraft is decreased. This problem will be referred to as the Aircraft Trajectory Optimization Problem (ATOP).

The purpose of this thesis project is to develop an optimization tool that is able to simultaneously optimize the arrival sequence and the corresponding approach trajectories of approaching aircraft in a dynamic optimization environment. This simultaneous optimization of the ASP and the ATOP requires that scheduling decisions are based on the performance data of computed achievable fuel-optimal approach trajectories. The approach trajectories must be free of conflicts and apply continuous descent procedures within the limits of existing approach procedure designs. By use of this tool, the goal is to continuously compute fuel-optimal arrival schedules based on the current required runway throughput.
Prior research has extensively dealt with solving the ASP and the ATOP statically, which means that only the currently available information on TMA entry times and aircraft types is considered while the effect of newly incoming aircraft on the arrival schedule is neglected. Unfortunately this static approach thus neglects any runway capacity related issues since it does not matter when the last aircraft lands. Some researchers have attempted to solve the dynamic approach problem however due to the computational intensity the ASP was either decoupled from the ATOP or the trajectory optimization was radically simplified by choosing between a few pre-defined approach trajectories instead of optimizing the trajectory specifically for each individual aircraft. Therefore, the research gap that this thesis project will attempt to fill is the dynamic and combined optimization of the arrival schedule and the approach trajectories. The designed tool can then be used to

compare the fuel consumption at various runway throughput levels to the currently attained performance using the FCFS scheduling strategy. Also an estimation of the maximum achievable runway capacity can be made.

The structure of this report is as follows. Chapter two contains all background information on the topic of aircraft arrival scheduling and approach trajectory optimization. This chapter will explain currently used approach operations and procedures, previous research done on both arrival sequencing and trajectory optimization and will finally discuss the optimization methods that are most commonly used to solve either the ASP or the ATOP and analyse previously used strategies to optimize the combination of both problems. In chapter three the thesis project is described in more detail by explaining the research gap and research objective. Also the research questions that will be answered in this report are stated. Chapter four will give a detailed explanation on the implementation of the optimization tool and the setup of an approach test case. Chapter five will present the results that will explain certain tool design choices made in chapter four and more importantly allow for a performance comparison with the currently used FCFS scheduling strategy. Finally, chapter six holds the conclusions that can be drawn from the designed optimization tool while chapter seven contains recommendations on how unanswered questions by this thesis research can be answered and what aspects of the aircraft approach optimization could be interesting for future research.

2

Background

The purpose of this chapter is to get an accurate overview of current practices and future concepts on the topic of Aircraft Sequencing and Trajectory Optimization and to discuss the required background information in order to fully understand the aircraft arrival problem.

Section 2.1 will provide the necessary background information on operations within the Terminal Maneuvering Area (TMA), which will help in identifying the constraints and boundary conditions of the aircraft arrival problem. Next sections 2.2 and 2.3 will respectively discuss the Aircraft Trajectory Optimization Problem (ATOP) and the Aircraft Sequencing Problem (ASP) which together form the actual optimization problem that is to be solved during this project. Lastly, section 2.4 will discuss possible optimization techniques and strategies that can be used for this optimization problem.

2.1. Terminal Maneuvering Area Operations

The terminal maneuvering area (TMA, or terminal control area TCA in the U.S. and Canada) is a control area that is established at high volume traffic airports that provides an IFR control service to arriving, departing and en route aircraft. Generally, the TMA airspace is designed to be circular in shape with the airport located in its center. The radius of the TMA is altitude dependent, usually ranging from 12 NM at 1200 ft AGL to 45 NM at and above 9500 ft AGL. In some cases, due to extremely high-volume traffic or a very complex structure of the airport's runway system, the TMA may be sectorized in order to gain an operational advantage. Any aircraft operating within the TMA are required to meet certain operating rules and equipment requirements [10].

2.1.1. TMA Entrance at Schiphol Airport

Currently, all approaching aircraft have to enter the TMA through one of several predefined approach corridors, depending on the direction from which they approach the TMA. Approach corridors are used to separate approaching aircraft from departing aircraft that leave the TMA via departure corridors [1]. The number of approach- and departure corridors varies for different airports. An example can be found in figure 2.1, showing the corridor system of Schiphol Airport.

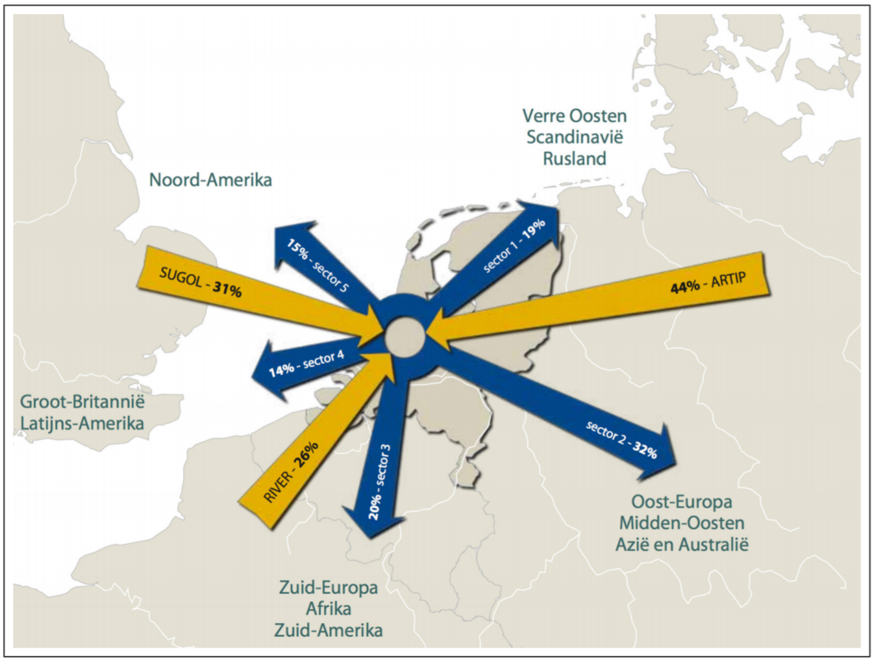


Figure 2.1: Approach- and departure corridors around Amsterdam Schiphol Airport [1]

Schiphol Airport uses three approach corridors for inbound traffic (SUGOL, RIVER and ARTIP) and five departure corridors for outbound traffic. The percentages represent the fraction of traffic that leaves or enters the TMA through the respective corridor. Arrival aircraft within an approach corridor follow standard terminal arrival routes (STARs) up to a common merge point where the aircraft are pre-sequenced, forming an arrival flow (or arrival stream) [11]. At this merge point the arrival aircraft are assigned with an available runway [1]. If necessary, the three arrival flows resulting from the approach corridors are merged and sequenced before the final approach fix (FAF).

2.1.2. RADAR VECTORING

The majority of todays TMAs use open loop radar vectors (heading instructions) for the merging of arrival flows [12]. The ATC radar units are able to detect the position of arrivals, which transmit their corresponding altitude information via transponders. The controller will then assign heading, altitude and speed to guide arrivals in his responsibility area.

A typical example is depicted in figure 2.2 below, where the radar vectoring technique is implemented to two incoming flows. The downwind flow requires three direction changes before the final runway approach while the base flow needs only two direction changes. All direction changes can vary, bounded between a maximum bearing angle and a minimum bearing angle, the latter of which usually represents the shortest route down to the runway. These variable angles allow the controller to lengthen or shorten the route of arrivals to ensure an optimal approach sequence when merging the two arrival streams. This is called path stretching or path shortening. Note that the initial vectoring location is not constrained to a particular point as shown in figure 2.2 but can theoretically start much sooner. The two turning axes are fixed.

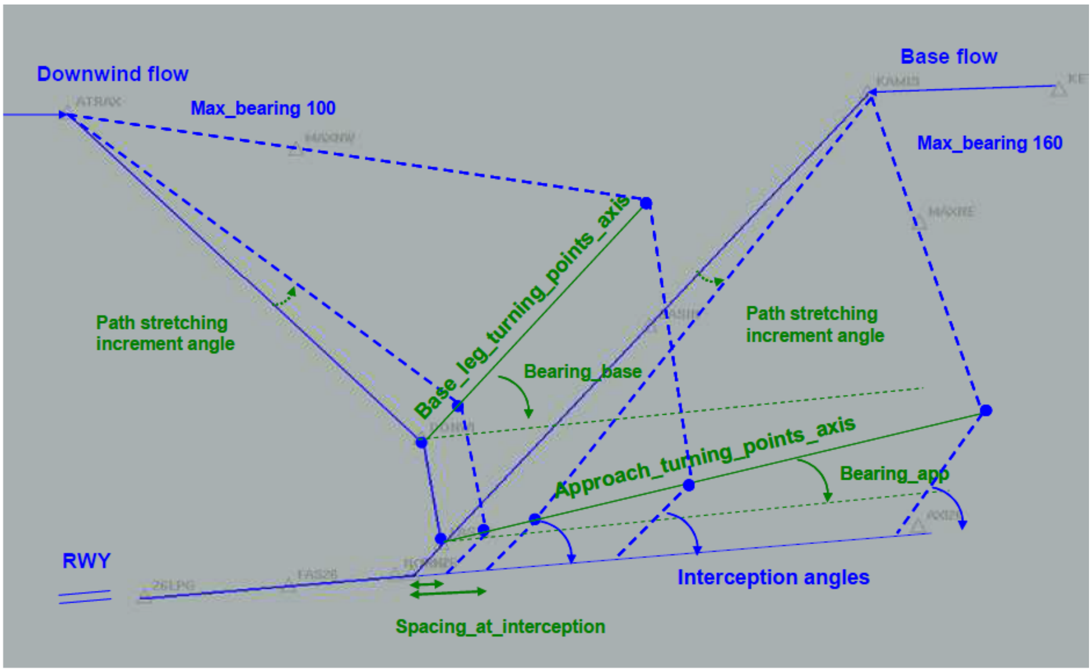


Figure 2.2: Vectoring method – path stretching with a given increment angle [2]

In order to avoid overloading this final approach system during peak traffic hours, airborne holding can be required. Airborne holding is implemented by having arrival aircraft fly an oval shaped pattern while waiting for a spot in the landing queue [13]. These airborne holding areas are often located next to the merge point, at the end of an approach corridor. A detailed standard arrival chart of Schiphol Airport, showing these holding locations together with the approach routes within the three approach corridors, can be found in Appendix A.3.

2.1.3. CONTINUOUS DESCENT APPROACH AT SCHIPHOL AIRPORT

During night-time, when air traffic is not even close to the maximum capacity, Schiphol Airport switches to using a continuous descent approach (CDA) procedure since this has been shown to provide significant reductions in time, fuel burn, emissions and noise impact [14]. Rather than descending down to the runway threshold in the conventional stairstep approach, where aircraft descend from level flight to level flight at a new lower altitude, the CDA procedure employs a constant-angle descent down to the glide slope intersection, where aircraft line up in front of the runway at a merge point located much further away from the airport. This also constraints the horizontal dispersion of arrival flight paths which further reduces the noise impact of the CDA procedure [2]. A conceptual representation of this procedure can be found in figure 2.4 of section 2.2, where more on CDA trajectories will be explained.

The reason for not using the CDA procedure during the much busier daytime hours is mostly the higher difficulty of arrival merging and sequencing, limited by the capabilities of air traffic controllers. This difficulty is caused by the reduced predictability of an aircraft's future position due to significantly varying speeds, compared to the conventional staged approach, where aircraft maintain the same speed and altitude at every stage [14].

2.2. TRAJECTORY OPTIMIZATION & CONFLICT RESOLUTION

In today's airports, TMA operations are largely restricted by limits in controller workload and communication arrival and departure aircraft. For this reason, increasing predictability of approach trajectories is the first priority. Unfortunately, this significantly limits the fuel-efficiency of current approach trajectories as aircraft cannot fly their preferred trajectories.

The value of trajectory optimization is very dependent on the accuracy of the information upon which the optimization is based. Aircraft following their preferred approach paths would limit the controller's knowledge of the intentions of the pilot and thus add a lot of uncertainty to the controller's trajectory predictions. In order to protect themselves from the risks of unforeseen events, controllers rely on standard procedures and fixed approach route structures with sufficient safety margins [15]. For these reasons, currently the optimizational aspect of approach trajectories mainly lies in the design of efficient approach route structures and approach concepts, rather than individually optimizing the trajectory of every single arrival aircraft as they enter the TMA.

However, there are significant potential benefits to be gained from optimizing individual aircraft's approach trajectories. This section will discuss optimal approach trajectories and will be explain what requirements an approach trajectory has to meet in order to be defined as being optimal. Also, frequently used terms in trajectory control will be explained. Further, future concepts aiming to improve currently used approach trajectories will be discussed.

2.2.1. BASIC PREMISES

Optimal Trajectories. Apart from critical requirements that an aircraft's trajectory has to satisfy, such as being free of conflicts, the optimality of a trajectory is mostly dependent on the objective that is given priority. For en-route aircraft, generally the aim is to find a trajectory that requires the least amount of fuel consumption. However, for approach trajectories many more factors can be part of the objective. The reduction of noise impact is one of the most important objectives for aircraft's approach trajectories. Also an approach trajectory's final conditions in terms of time and position are usually fixed in order to comply with a landing sequence, imposed by the controller.

Conflicts. A conflict is defined as the loss of the minimum allowed separation distance between two aircraft [16]. The minimum separation distance is dependent on the class of airspace and is defined in horizontal and vertical direction. In the TMA, the vertical minimum separation is 305 meters (1000ft) and the horizontal minimum separation is 5.56 kilometres (3NM), as defined by the ICAO [17]. These separations distances combined form a disk shaped volume around an aircraft as can be seen in figure 2.3. Aircraft in the TMA that are following the same approach path have to follow wake turbulence separation requirements, as will be discussed in section 2.3.1, which will then take the place of the horizontal separation requirements.

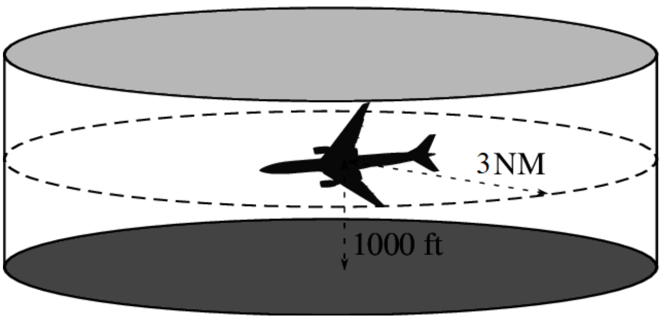


Figure 2.3: A conceptual representation of the conventional step descent and the CDA

Cooperative and non-cooperative conflict resolution. A conflict between two (or more) aircraft trajectories is generally resolved by one of two possible methods. Non-cooperative conflict resolution assumes there is no exchange of information between the two conflicting aircraft. The ATC will route one of the two aircraft in such a way that the conflict is resolved and no other conflict is created. At the same time, the other aircraft will stay on its original trajectory and does not take part in the resolution of the conflict.
Cooperative conflict resolution requires both aircraft to take part in the resolution of the conflict. The ATC will route both aircraft around each other, after which they will continue on their original trajectory. Currently, the non-cooperative method is used since this method is less intensive in terms of controller workload. The cooperative method however offers far greater optimization potential and is therefore typically formulated as an optimization problem. The ATC can design the flight paths such that conflict are avoided while a the same time a certain cost function is minimized[18].

Tactical and strategic trajectory optimization. Optimizing and de-conflicting trajectories is generally done using either a tactical or a strategic method. Strategic methods compute optimal trajectories on time scales of thirty minutes or more into the future [19]. On the contrary, tactical methods resolve conflicts when they arise, on time scales up to 15 minutes, without taking into account what future consequences this trajectory modification has. Unfortunately, current operational ATC procedures are of a tactical nature, which generally does not result in a globally fuel-efficient flight operation [15].

2.2.2. CONTINUOUS DESCENT APPROACH (CDA)

An approach procedure that has seen widespread adoption in today's airports is the continuous descent approach (CDA) that has been firstly developed by Erkelens [8] from the Dutch National Aerospace Laboratory NLR. Erkelens' main goal was to solve the noise problem around airports due to jet aircraft approaching at low altitude. In general, CDA procedures are vertically optimized approach trajectories [3]. Eurocontrol [20] defines the CDA procedure as: "an aircraft operating technique in which arrival aircraft descend from an optimal position with minimum thrusts and avoid level flight to the extent permitted by the safe operation of the aircraft and compliance with published procedures and ATC instructions." The main aim of the CDA is to reduce noise emissions, atmospheric emissions and fuel consumption.

In CDA approaches procedures two different types exist, a *tactical* CDA and a *advanced* CDA. The difference between the two comes down to their lateral path design. In a tactical CDA, the lateral path to follow is defined by the ATC, who provides the pilot with vectoring instructions. An advanced CDA follows a predefined Standard Terminal Arrival Route (STAR), which means that all arriving aircraft follow the same lateral approach path.

The goal of developing an airport CDA procedure is to keep approaching aircraft at higher altitudes for as long as possible and to keep the aircraft's thrust setting as low as possible during descent. An ideal CDA would allow the aircraft's thrust setting to be at idle during the majority of the descent [3]. The higher altitude approach profile allows for a more fuel efficient flight and reduces the noise experienced by residents living in the area. A conceptual representation of the CDA and a conventional step descent can be found in figure 2.4.

In a typical CDA procedure, the approaching aircraft starts descending from an Initial Approach Fix (IAF) at an altitude of approximately 8000ft to 10000ft and at around 25nm to 30nm before the runway. The Final Approach Fix (FAF) is reached at an altitude of around 2500ft, where the ILS is intercepted. From here, the aircraft descends to touchdown along a glide slope of three degrees [3]. The lateral flight path of the CDA procedure is fixed and usually designed to avoid highly populated areas close to the airport.

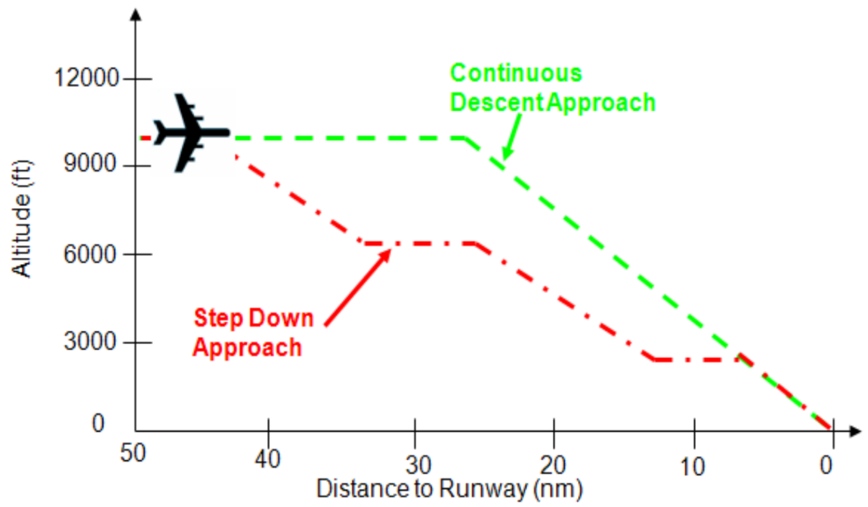


Figure 2.4: A conceptual representation of the conventional step descent and the CDA (from Alam 2010 [3])

The most significant drawback of the CDA procedure is the reduced predictability of an aircraft's future position [14]. Since a CDA is performed at idle thrust settings, an aircraft's trajectory depends not only on atmospherical conditions, such as the current wind speed and temperature, but also on the aircraft's individual performance characteristics [21]. Apart from this, during a CDA aircraft are continuously descending which causes trailing aircraft to close in on a leading aircraft since aircraft at higher altitude fly at a higher true airspeed (TAS). This effect of shrinking horizontal distance between approaching aircraft is often referred to as the "compression effect" [22].

As a consequence, the separation distance must be increased substantially. A typical separation distance of a CDA procedure is around four minutes [8]. All in all. this leads to increased controller workload due the the reduced predictability of the approach trajectories while at the same time the runway capacity is decreased due to the longer required separation distances [23]. For this reason, current CDA procedures can only be carried out during night hours, when the traffic density is substantially lower.

2.2.3. Trajectory Based Operations (TBO)

The lack of trajectory predictability when performing CDAs could potentially be solved by the use of Trajectory Based Operations (TBO). TBO makes use of 4-Dimensional (4DT) reference trajectories which means that the aircraft's latitude, longitude and altitude are known at any time along the planned trajectory. These reference trajectories are tailored specifically to the individual aircraft's performance characteristics. The goal of TBO is to fly the aircraft as closely as possible along the user-preferred trajectory [24].

Fundamental to the use of TBO is the ability of airborne equipment to provide closed-loop control to this reference trajectory, in order to accurately track the planned 4DT trajectory. Modern aircraft feature a Flight Management System (FMS) that can generate these 4DT trajectories based on crew-entered or uplinked data and the current aircraft state- and atmospheric data. Consequently, the aircraft's Automatic Flight Control System (AFCS) is able to control the aircraft along the trajectory generated by the FMS. Knowing the preferred trajectories of all aircraft in the area, conflicts can be solved strategically (i.e. ahead of time) via the negotiation of trajectory revisions [24].

In order to evaluate the Required Time of Arrival (RTA) function of a Boeing 737 FMS, GE Aviation Systems performed 33 trial fligths with Scandinavian Airlines in 2001. These flight trials demonstrated that aircraft equipped with the current generation FMS could reliably predict and maintain a 4DT trajectory over an entire flight. The accuracy at waypoints located at the top of the arrival procedures was measured to be less than 7 seconds with an average of around 4.8 seconds [25].

2.3. ARRIVAL SEQUENCING & RUNWAY CAPACITY

This section will discuss how arrival aircraft are currently sequenced into the landing queue before the FAF and how the arrival sequencing could be approached in the future. Further, the influence of arrival sequencing on the effective runway capacity will be explained. The problem of sequencing arrival aircraft will be referred to as the *arrival sequencing problem* (ASP).

2.3.1. BASIC PREMISES

Capacity. The capacity of an airport's single- or multi runway system is defined as the expected number of movements (landings of take-offs) that can be performed per unit of time, assuming continuous demand whilst maintaining the ATC separation requirements. This is also referred to as "maximum throughput capacity" [13]. According to Ball [13], single runway capacity, assuming no other runway dependencies such as crossings, depends on the following most significant factors:

- The aircraft weight classes that are using the runway
- The weight-class-dependent(wake vortex) separation requirements, imposed by regulation
- The runway operation type (arrivals only, departures only or mixed operations)
- The local weather conditions (visibility, wind, cloud ceiling and precipitation)
- The available technology and overall performance of the local ATM system

All of these factors, combined with the number of arrival aircraft demanding a spot in the landing queue, affect what landing sequence would be optimal with respect to delay or fuel efficiency. Important to keep in mind is that increased demand on available runway capacity will force the landing sequence to prioritize runway throughput over fuel efficiency or delay objectives. De Neufville [26] accurately explains this relation between capacity, demand and delay as follows: “The performance of a service system is, indeed, sensitive to the pattern of loads especially when they approach its capacity. The capacity of a service facility is, thus, not at all similar to our notion of capacity in everyday life, that is, the volume that a bottle or other vessel can hold. A bottle will accommodate any amount of liquid up to its capacity equally well; and after that, it can hold no more. A service facility, on the other hand, does not provide equal service at all times; its service rapidly deteriorates as traffic nears capacity. A service facility, can, furthermore, eventually handle more than its immediate capacity by delaying traffic until an opportunity for service exists."

Practical Hourly Capacity (PHCAP). The maximum throughput capacity can only be sustained for periods no longer than one or two hours, since delay would build up to unacceptable levels. For this reason, another definition of delay, called *"Practical Hourly Capacity"*, is frequently utilized. The practical hourly capacity (PHCAP) is defined as the expected number of movements that can be performed within a one hour period on a runway system, without exceeding an average delay of four minutes. The PHCAP is reached the moment this threshold is exceeded. Depending on the specific condition, generally the PHCAP is reached at around 80-90% of the maximum throughput capacity [11].

Wake Separation. Arriving aircraft operate at high angles of attack in order to generate enough lift at lower airspeeds. This flight attitude causes strong vortices trailing from each wingtip significantly increasing wake turbulence behind the aircraft. In order to ensure safe operations for all arrivals, situations in which other aircraft encounter such turbulence have to be prevented. Therefore, the ICAO has established mandatory separation minima, a minimum distance that a trailing aircraft must stay behind a leading aircraft in the arrival landing queue. These separation distances are based on the Maximum Take Off Weight (MTOW) since wake turbulence strength is primarily determined by aircraft weight and speed. The MTOW was originally divided into three classes (i.e. heavy, medium or light) that allocated all aircraft. However, since the separations are defined on the worst case in each category, this led to frequent over separation. To tackle this problem, in Europe a more precise wake vortex categorization (RECAT-EU) was brought out, dividing the "heavy" and "medium" categories into two sub-categories and even creating a new "super-heavy" category for the Airbus A380 and the Antonov AN-124. A table with all corresponding minimum separation distances between leading and following aircraft on approach and departure can be found in table 2.1 below. Due to these separation minima, some aircraft pairs require longer separation distances than others. Assuming the static ASP problem, this means that the total amount of time required to land a certain set of aircraft is dependent on the landing sequence of these aircraft. Note that when minimum separation distances are not required (e.g. for "upper heavy" following "upper medium), minimum radar separation MRS of 3NM applies in the TMA [5].

RECAT-EU scheme		"SUPER HEAVY"	"UPPER HEAVY"	"LOWER HEAVY"	"UPPER MEDIUM"	"LOWER MEDIUM"	"LIGHT"
Leader / Follower		"A"	"B"	"C"	"D"	"E"	"F"
"SUPER HEAVY"	"A"	3 NM	4 NM	5 NM	5 NM	6 NM	8 NM
"UPPER HEAVY"	"B"		3 NM	4 NM	4 NM	5 NM	7 NM
"LOWER HEAVY"	"C"		(*)	3 NM	3 NM	4 NM	6 NM
"UPPER MEDIUM"	"D"						5 NM
"LOWER MEDIUM"	"E"						4 NM
"LIGHT"	"F"						3 NM

Table 2.1: RECAT-EU, wake turbulence separation distance minima on approach and departure, from Rooseleer [5]

First-Come-First-Served (FCFS). Currently two systems, TMA (Traffic Management Advisor) and FAST (Final Approach Spacing Tool), are used by ATC for the sequencing and spacing of incoming aircraft. Both systems sequence incoming aircraft based on the so called "first-come-first-served" schedule [27]. In the FCFS schedule, incoming aircraft are scheduled in order of their scheduled runway arrival times. Key advantages of the FCFS sequence are reduced controller workload due to the easy implementation of this schedule and maintaining a sense of fairness amongst different aircraft classes since each incoming aircraft is treated equal [28]. However, it has been found that FCFS is rarely the most optimal sequencing order in terms of runway capacity or average delay due to large separation requirements caused by randomly mixed aircraft weight classes.

Grouping. In a multiple runway system it is possible to optimize operations by judiciously assigning operations or aircraft classes to different runways [13]. Assuming a case of two sufficiently spaced parallel runways, it could be beneficial to use one runway solely for arrivals and using the other solely for departures. This allocation strategy reduces controller workload since arrivals and departures can operate independently.

In another scenario where two or more runways are used only for arrivals, ATMs generally try to assign homogeneous mixes of aircraft classes to each of the runways as much as possible. Using this strategy, often referred to as "grouping", controllers can more easily avoid the large wake-vortex separations that are required whenever a light aircraft is queued up behind a heavy aircraft.

2.3.2. TIME-BASED WAKE SEPARATION

As discussed, arrival aircraft are required to maintain minimum wake separation distances when descending from the runway threshold down to the runway. These separation distances are solely based on the weight classes of a pair of subsequent aircraft. Although this separation rule has been used for many years, it is not the most consistent way of defining separation.

Headwind conditions on final approach cause a ground speed reduction of the approaching aircraft. Due to the reduced ground speed, the aircraft requires more time to cover the separation distance to the preceding aircraft which causes the runway to be unoccupied longer which reduces the runway's landing rate [29]. In a similar fashion, aircraft with slower approach speeds have the same effect on a runway's landing rate.

SESAR JU [30] has come up with a concept called *"time-based separation"* which replaces current distance separations with time intervals in order to provide consistent time-based spacing and maintain the runway's effective landing capacity. This time-based separation concept has been validated in the framework of SESAR Development by Moris and Peter [29]. The assessment concluded that the time-based separation concept is viable from a perspective of safety and human performance while allowing for significant benefits in terms of recovering the landing rate reduction in headwind conditions. A table with the corresponding time spacings between leading and following aircraft weight classes can be found in figure 2.2 Currently, the procedure is in daily use at London Heathrow airport, where in strong wind conditions up to five additional aircraft landings are possible compared to the traditional time-based separation procedure [30].

RECAT-EU scheme		"Super Heavy"	"Upper Heavy"	"Lower Heavy"	"Upper Medium"	"Lower Medium"	"Light"
Leader / Follower		"A"	"B"	"C"	"D"	"E"	"F"
"Super Heavy"	"A"		**100s**	**120s**	**140s**	**160s**	**180s**
"Upper Heavy"	"B"				**100s**	**120s**	**140s**
"Lower Heavy"	"C"				**80s**	**100s**	**120s**
"Upper Medium"	"D"						**120s**
"Lower Medium"	"E"						**100s**
"Light"	"F"						**80s**

Table 2.2: RECAT-EU, time-based wake turbulence separation minima on approach and departure, from Rooseleer [5]

2.3.3. CONSTRAINED POSITION SHIFTING (CPS)

Dear and Sherif [6] have come up with a concept called constrained position shifting (CPS) which deviates from the commonly used FCFS strategy by allowing arriving aircraft to shift a limited number of positions forward or backward in the arrivals queue in case this would lead to a more optimal landing sequence. This means that if an arriving aircraft would have to land in the 9th position (assuming the FCFS strategy), this aircraft could now be scheduled anywhere within positions 7-11, assuming the maximum position shift was set at 2. This strategy was tested using a static single runway case of a set of around 500 aircraft to simulate extreme situations of approximately 11-12 hours of continuous peak period arrival congestion. These aircraft were of different weight- and velocity classes. Demanded runway throughput rates near, at and over the maximum runway capacity were tested, which was 35, 40 and 45 arrivals per hour in this case. With the maximum position shift set to 4, this resulted in a 16-67% reduction of average delay compared to the FCFS strategy. Interesting is the fact the potential reduction of average delay significantly increases for higher demanded runway throughputs, as can be seen in table 2.3 below. Venkatakrishnan [31] adopted the CPS approach and applied it to a more realistic multiple runway configuration. The ASP was modelled as a dynamic problem by reconsidering the landing sequence every time a new arrival enters the TMA area. It was found that delays could be reduced up to 30% using sequences suggested by the algorithm, compared to the FCFS strategy commonly used by ATCs.

		Arrivals Per Hour		
		35	40	45
Case 1	**FCFS-RW**	**183**	**381**	**1689**
(Capacity = 40.8h^{-1})	**CPS**	**153**	**300**	**957**
	% Reduction	**16.4**	**21.3**	**43.5**
Case 2	**FCFS-RW**	**357**	**1401**	
(Capacity = 38.1h^{-1})	**CPS**	**274**	**550**	
	% Reduction	**23.2**	**60.7**	
Case 3	**FCFS-RW**	**130**	**322**	**963**
(Capacity = 45.5h^{-1})	**CPS**	**75**	**121**	**319**
	% Reduction	**42.3**	**62.4**	**66.9**

Table 2.3: Average delay (sec) comparison between FCFS and CPS strategies [6]

2.3.4. THE STATIC & DYNAMIC ASP MODEL

The aircraft sequencing problem (ASP) has been extensively researched since the 1980s. Due to the very complex matter of efficiently using available runway capacity by optimal arrival sequencing, it is evident that an automated solution to the ASP is the way to go. It is important to identify that the ASP can be modelled either as a static or dynamic problem [32].

The static ASP model assumes that all information, such as the arrival TMA entrance times, the number of arrival aircraft, their speeds and their aircraft types is known in advance, before the arrivals are sequenced. Mostly, the optimization objective will be to minimize the time it takes to land the set of arrival aircraft. In a static environment there is no constraint on the time at which the last aircraft lands.

In the dynamic ASP model, aircraft enter the TMA one at a time. This information is not known in advance, meaning that a new landing sequence has to be recomputed every time a new aircraft is added to the arrival aircraft pool. The difference to the static model can sometimes be quite artificial if one assumes that any aircraft that has already been scheduled before can be resequenced whenever a new arrival enters the TMA. This would mean that a static ASP algorithm could be used over and over, whenever a new aircraft enters the TMA and would then equal a dynamic ASP algorithm.

Although static cases can be useful to test different optimization methods and strategies, they lack a very important aspect of the ASP, namely the variation in demanded runway throughput in combination with a limited runway capacity. In a static environment there is no constraint on the time at which the last aircraft lands, which means that delays do not propagate on to new arrival aircraft entering the TMA. In reality, the ASP is a very dynamic problem. In a dynamic environment, this last arrival time significantly affects the landing times of new aircraft entering the TMA. The number of aircraft waiting to land changes continuously as some aircraft reach the runway while others join the landing queue. This also affects the objective function significantly, as minimizing the "latest landing time" makes no sense any longer. Objective functions, such as minimizing the average passenger waiting time would make more sense [13].

2.3.5. Freeze horizon & rolling horizon scheduling concepts

Dynamic models generally tend to confine and scale down the total ASP, aiming to speed up computation times allowing for real-time optimization of the ASP. Murca et al. [33] mentions that generally, dynamic aircraft sequencing is addressed through concepts, such as freeze horizon and rolling horizon (or receding horizon control).

The freeze horizon concepts generally use two regions that are centered around the runway threshold, the scheduling horizon (SH) and the innermost freeze horizon (FH). Any aircraft entering the SH region will be entered into the aircraft scheduling and sequencing process. Whenever a new aircraft enters the SH region, all aircraft that are still within the SH region can be resequenced. Once an aircraft leave the SH and enter the innermost FH region, its position in the landing sequence is "frozen". This concept can be seen as a combination of cooperative aircraft scheduling (within the SH) and non-cooperative aircraft scheduling (within the FH).

The rolling horizon concept splits the ASP up into a number of time intervals that are optimized individually. After solving the ASP for one interval, the solution of time-dependent variables is passed on to the following interval. An example of this concept can be found in figure 2.5.

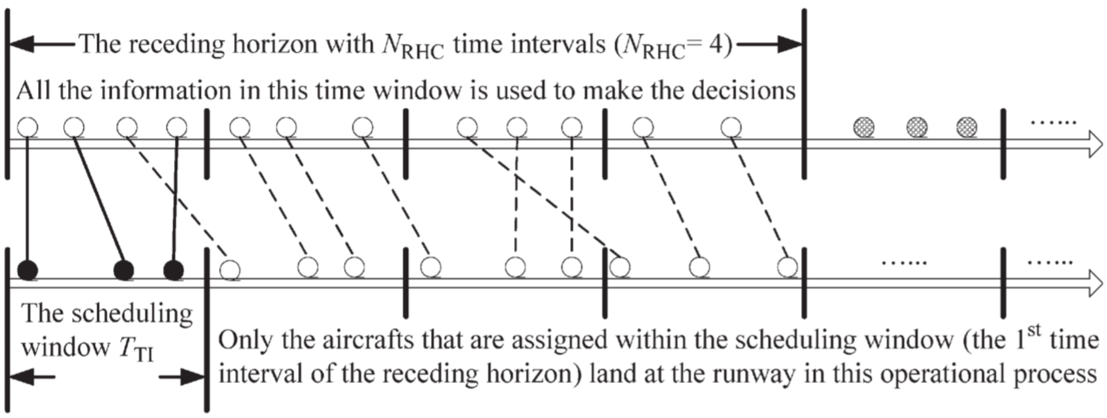


Figure 2.5: Example of the rolling horizon control concept (from Zhan [4])

2.4. Optimization Methods & Strategies

The optimization of aircraft approach and landing can generally be divided into two sub-optimizations. Firstly, it is important to determine the optimal landing sequence in which aircraft arrive at the runway threshold, considering the mandatory separation requirements. Secondly, the optimum aircraft trajectories in order to fulfil the established landing sequence need to be computed, while resolving any arising trajectory conflicts.

Mathematically, trajectory optimization is a continuous problem while aircraft sequencing is a discrete problem. Therefore, although the aircraft sequencing problem (ASP) and aircraft trajectory optimization problem (ATOP) are heavily dependent on each other (any time a landing sequence is changed, entirely new optimal trajectories need to be computed), the two problems require two very different methods to be solved efficiently. This makes it difficult to solve both problems simultaneously.

This section will explain how the simultaneous optimization of these two problems has been solved previously and what problems can be expected. Before doing so, the actual optimization objectives will be clarified.

2.4.1. Objectives and Quantification

Generally, the optimization of aircraft approach and landing is aimed at reducing flight delay and reducing fuel consumption. Both optimization problems, arrival sequencing and trajectory optimization, play a different role with respect to these objectives. Usually, arrival sequencing is the leading problem that generates the final conditions (i.e. a landing sequence) that are used for the trajectory optimization problem. As a result, in most cases the ASP is aimed at reducing delay while the trajectories are optimized for fuel-efficiency.

Flight delay can be defined in numerous ways. Since this thesis deals purely with the arrival phase of a flight, only delay resulting from terminal area operations will be considered. For every individual aircraft a fuel-optimal approach trajectory can be computed assuming that the aircraft can land independently of other traffic in the area. The landing time that results from this fuel-optimal approach trajectory is considered the target time of arrival (TTA). The difference between the aircraft's TTA and its actual arrival time is a measure of how much the aircraft is delayed [31]. The following quantifications of the cost of delays are most commonly found:

- The sum of the time delay suffered by all aircraft in the considered set [31].
- The sum of the time delay suffered by all passengers in the set of considered aircraft. This quantification prioritizes aircraft carrying more passengers [31].
- The maximum individual delay of an aircraft in the considered set [34].
- The total monetary cost (e.g. from fuel consumption or mechanical wear) due to delaying aircraft in the considered set [31].

The cost of fuel consumption is usually quantified as the cumulative fuel consumed by all aircraft in the considered set. This is because fuel consumption has no significant impact on further airport operations such as flight delay, which often propagates on to the next flight.

Lastly, a relevant optimization objective can be maximum throughput [6]. In a static optimization case this can simply be quantified as the time at which the last aircraft in the considered set lands. In a dynamic environment, runway throughput is quantified as the number of aircraft landings per unit of time.

2.4.2. Optimization Methods

Before covering how the ASP and ATOP can be solved simultaneously, firstly the conventional methods of optimizing these two problems individually will be discussed. Various methods and strategies have been investigated in previous works. The used methods can generally be divided into two types, numerical methods and heuristic methods. Numerical or calculus-based methods approach the trajectory optimization problem with a definite convergence criterion whereas heuristic methods randomly select values for unknown problem variables, after which the best answer is considered to be the answer [35]. Heuristic methods often discretize the possible solutions of the continuous trajectory problem.

Trajectory Optimization. Betts [35] presented a survey of trajectory optimization methods. He states that discrete problems, such as aircraft sequencing, are usually solved using heuristic methods (e.g. genetic algorithms, simulated annealing, tabu search, and evolutionary or Monte Carlo methods). The reason for this is that for discrete problems it is impossible to define convergence which makes it difficult to properly define termination criteria. The only way to determine whether a candidate solution $\hat{x}$ is an optimum solution is by comparison with every other possible candidate.

Trajectory control problems, on the other hand, can be solved using calculus-based methods. Betts recommends using numerical, gradient-based optimization methods such as direct shooting, indirect multiple shooting, and direct transcription to solve optimum control problems and explains that discrete methods are not computationally competitive in solving trajectory problems as they do not exploit available gradient information.
Medioni et al. [36] agrees that gradient-based optimization methods have a significant performance advantage over heuristic methods in terms of providing a quick and efficient solution to a trajectory control problem. However, he also states that gradient-based methods tend to converge to the nearest local minimum and therefore often do not find the globally optimal solution. Especially in complex trajectory control problems, with many aircraft involved, this tends to be a problem to consider.

Aircraft Sequencing. The ASP is a discrete optimization problem since its cost function that is to be minimized is discontinuous and therefore not differentiable over the entire sequencing interval. For this reason, aircraft sequencing is usually solved using heuristic optimization methods. Genetic algorithms are popular due to their ease of use and seem to be promising, especially for more complex problems with larger numbers of arrival aircraft.
Hansen [37] investigated the effectiveness and efficiency of genetic algorithms for the routing and sequencing of arriving aircraft in a multiple runway system. Hansen reported that real time near-optimal solutions were achieved for problems of a realistic size. Especially the scalability to more complex problems with many arrival aircraft was reported to be a major advantage of genetic algorithms.
Another method that has often been used to solve the ASP for minimum delay is mixed integer programming. Artiouchine [38] developed a mixed integer programming formulation that allows to solve the general static single runway ASP for a large number of aircraft implementing holding patterns. This formulation was able to minimize the number of times a plane entered the holding pattern, given a minimum time separation between consecutive landings.
The downside of using mixed integer programming is that it is much harder to solve the ASP for minimum-fuel, since discrete sequencing is now combined with continuous problem of scheduling the actual landing times.

2.4.3. SIMULTANEOUS OPTIMIZATION

Since optimal control problems use a gradient-based optimization routine, it is most likely that the arrival sequence that was used as an initial guess (or starting point of an optimization) will remain unchanged. The reason for this is that the solver would have to route two or more arrivals around each other in order to change their approach sequence. Doing so would significantly increase the cost function at first before the solver arrives at a new sequence and hopefully at a more optimum solution. Due to this initial increase in the cost function, the solver will falsely assume that this is not the “right way to go” and stick to the original approach sequence.

Hansen [37] was one of the first to combine the ASP with aircraft approach routing optimization. As also mentioned in section 2.4.2, he investigated the effectiveness and efficiency of genetic algorithms for the routing and sequencing of arriving aircraft in a multiple runway system. A significant drawback of this optimization strategy was that the approach routing was modelled as a choice between a number of predefined discretized approach routes. This means that the algorithm would choose the best available approach trajectory instead of actually optimizing the trajectory itself.

Toratani et al. [39] developed a hybrid optimization algorithm to simultaneously solve the ASP and ATOP for a 4-aircraft static case, using a single runway with two possible entry points. The objective was to find fuel optimal trajectories for all aircraft. Firstly, a mixed integer linear programming (MILP) method was used to find the optimal TMA entry point and place in the landing queue. Consequently, an optimal control method was employed to find fuel-optimal trajectories for each aircraft independently (assuming there are no other aircraft in the vicinity). The calculated arrival time was then denoted as the TTA. Lastly, these TTAs were used in the objective function aiming to minimize all aircraft's total deviation from their individual TTA.

Sàma et al. [7] proposed a framework for the lexicographic optimization of both, the Arrival Scheduling Problem (ASP) and the Aircraft Trajectory Optimization Problem (ATOP). In this paper the ASP was modelled as a job shop scheduling problem with additional constraints and the ATOP was formulated as an optimal control problem. Sàma explains that the goal of the proposed framework is to solve both problems sequentially based on a defined lexicographic order of importance for the performance indicators, where to most important performance indicator defines the first problem to be solved. In this paper three performance indicators were evaluated, namely total fuel consumption, maximum individual delay and total travel time. Two alternative approaches were investigated. The first approach was to solve the ATOP before the ASP. The optimal trajectory for each aircraft would be computed either for minimized travel time of for minimized fuel-consumption. Next, these trajectories would be used to solve the ASP. The other approach would be the opposite. The ASP would be solved to minimize the individual aircraft delay after which the ATOP would be solved, again for either minimum travel time or minimum fuel consumption. Both optimization approaches were tested on real-world data. The results can be found in table 2.4. Interesting to see is that for any of the four approaches the total travel time does not change significantly while the maximum individual delay increases drastically whenever the ASP is not solved first. The minimum fuel consumption is achieved when the ATOP is solved for minimum fuel first after which the ASP is solved. Although the resulted fuel consumption is much lower than for any of the other approaches, the cost in delay and travel time is very significant.

First Problem	Second Problem	Maximum Delay	Travel Time	Fuel Consumption
TCA-ASP	ATOP-tt	**59**	10582	2174
TCA-ASP	ATOP-fc	**59**	11962	1288
ATOP-tt	TCA-ASP	262	**10177**	2433
ATOP-fc	TCA-ASP	501	13637	**276**

Table 2.4: Results of various lexicographic approaches on real-world data [7]

3
Research Objective

Based on the relevant background information given in previous chapter, this chapter will elaborate on the research objective and research questions of this thesis research. Before doing so, the research gap in academic literature on the topic of aircraft sequencing and approach trajectory optimization will be identified.

3.1. Research Gap

Existing literature mainly deals with combining the aircraft sequencing problem (ASP) with the aircraft trajectory optimization problem (ATOP) for static aircraft approach cases. A few researches have also solved the ASP in a dynamic optimization environment, isolating the ASP from trajectory optimization.

However, a research gap was found in the combined optimization of the ASP and the ATOP in a dynamic optimization environment. A dynamic environment also considers the aspect of limited capacity and takes into account that not all arrival information is known upfront. Apart from this not only the boundary conditions (entry- and landing conditions of arrival aircraft) are dynamic but also the optimization objectives must be made dynamic. For example, it makes no sense at all to optimize for maximum runway throughput when the levels of traffic volume are far below the maximum capacity.

3.2. Research Objective

The thesis objective is to develop a hybrid optimization tool that is able to solve the ASP and to simultaneously find optimal trajectories to merge arrival aircraft in the found sequence, applying advanced CDA procedures within the limits of existing STAR or approach procedure designs. The latter means that overtaking, to apply a landing sequence, will take place in the vertical plane while following an identical horizontal approach path (e.g. an existing STAR). The algorithm must be able to handle a dynamic TMA environment and account for a variable demanded throughput of a single-runway system. When completed, the algorithm can be used to determine the sensitivity of fuel- and delay objectives to several input parameters such as the demanded runway system throughput in consequence of varying levels of traffic volume.

3.3. Research Questions

Based on the research gap and objective, the main research question is defined as follows:

How to balance fuel-optimality and time-optimality objectives in a dynamic scheduling environment for a given demanded runway throughput, based on the combined optimization of arrival sequences and approach trajectories, applying continuous descent operations?

The main research question is supported by and based on the following set of subquestions:

- Individual Approach Trajectory Optimization
  - What is an aircraft's earliest time of arrival and what is its trajectory?
  - What is an aircraft's latest time of arrival and what is its trajectory?
  - What is an aircraft's fuel-optimal time of arrival and its trajectory neglecting trajectory conflicts?
- Aircraft Scheduling
  - What landing sequence maximizes the runway system throughput?
  - What landing sequence minimizes either the maximum individual delay, the total delay, or the total fuel consumption?
  - What aircraft sequencing strategy can be used in order to optimize fuel- or delay based objectives, assuming a user-defined minimum runway throughput?
  - When can a landing sequence solution be considered achievable?
  - How much do arrival scheduling concepts, such as freeze horizon,rolling horizon and constrained position shifting, affect the results of the scheduling objectives?
- Conflict Resolution
  - How and when can trajectory conflicts be detected?
  - How can possible trajectory conflicts be resolved?
- Test Case Setup
  - How to model a single runway system in a dynamic TMA environment?
  - How to set up initial conditions for a dynamic TMA environment?
- Optimization Objectives
  - What is the theoretical limit of runway throughput and what factors affect this?
  - How does the demanded runway system throughput affect the fuel consumption of a set of arrival aircraft?
  - How does the demanded runway system throughput affect the individual delay of a set of arrival aircraft?
  - How does the demanded runway system throughput affect the total delay of a set of arrival aircraft?
  - Using dynamic optimization objectives, at what demanded throughput threshold should the optimization objective be switched between minimizing fuel consumption and maximizing runway throughput?
  - In a dynamic environment, does the minimum fuel sequencing objective actually yield minimized fuel consumption over the long term?

4

IMPLEMENTATION

4.1. REQUIREMENTS

First and foremost, before deciding on what optimization strategy can be used to solve the aircraft approach problem, the necessary requirements must be defined that the optimization tool has to meet in order to give a decisive conclusion to the research question and to close the identified research gap. The most critical requirements are listed below.

- The optimization tool must be able to simultaneously optimize the ASP and ATOP for a single runway system.
- The optimization tool must be able to detect and resolve possible trajectory conflicts using minimal fuel while still meeting the target time of arrival.
- The optimization tool must be able to handle a dynamic environment where no prior knowledge of incoming aircraft is available.
- The optimization tool must be able to automatically switch between different optimization objectives based on varying levels of required runway throughput.
- The optimization tool must be able to apply CDA procedures within the limits of existing STAR designs.

4.2. OPTIMIZATION OBJECTIVES

For the optimization of this aircraft approach problem three different optimization objectives will be defined:

- **Minimum Fuel Consumption:** The fuel consumption objective is defined as the total amount of fuel burned by all approaching aircraft. An option of using normalized fuel curves will be added to maintain a sense of fairness between all aircraft weight classes and to determine what effect this has on the resulting sequences and trajectories.

- **Minimum Time (Maximum Throughput):** The minimum time objective will aim to minimize the time of arrival of the last aircraft in the considered set. This means that the set of aircraft will land in the smallest possible time interval. Therefore, this objective can also be referred to as maximum throughput.

- **Minimum Delay:** Delay is defined as being the difference between the fuel-optimal landing time and the actual landing time. If the actual landing time is earlier than the fuel-optimal landing time, the delay is counted as zero. Since this optimization only considers the approach phase of a flight, accumulated delay during earlier flight phases will be neglected.

4.3. INITIAL CONDITIONS

Lastly, before diving into the actual aircraft approach optimization, the initial conditions (i.e. the TMA entry conditions of all approaching aircraft) for each approaching aircraft must to be defined:

- **Entry Time**: the time at which an aircraft enters the final approach route (the chosen approach route is given in appendices A.1 and A.2 and is explained in section 4.5.2).
- **Aircraft Type**: The type of aircraft and more importantly the aircraft's weight class.

Since the optimization problem is of a dynamic kind, the optimization tool has no knowledge of the entry conditions of all approaching aircraft beforehand as it would be the case when optimizing a static approach case with a fixed pool of approaching aircraft. Two options of generating dynamic initial conditions seem to be feasible:

- Generating the required entry conditions "on the spot" while the optimization process is already running.
- Generating a pool of aircraft, defining all individual entry conditions before the optimization is started but sharing the individual entry conditions only at pre-determined times at which this information would realistically become accessible.

The latter of these two options has been chosen because this enables the user to run the exact same approach problem multiple times, making it easier to see how changes within the optimization routine affect the resulting sequences and trajectories. The entry times for a dynamic approach case are determined using the following steps:

1. The user defines the demanded throughput [*aircraft entries per hour*].
2. The user defines the size of the aircraft pool [*number of aircraft*].
3. From this information the duration of the approach case can be calculated (e.g. for a pool of 100 aircraft and a demanded throughput of 48 AC/hr, the time interval during which all aircraft enter the final approach route is 125 minutes).
4. Within the previously determined time interval, entrance times are generated for every aircraft in the pool, based on a continuous uniform distribution. The only requirement is that basic separation distances apply. Therefore the minimum time interval δ_t between two TMA entrances is limited by the minimum radar separation and the true airspeed at the entrance:

$$\delta_{tmin} = \frac{MRS}{V_{max}} = \frac{3.0NM}{250KIAS_{@FL100}} \approx 37.41s \tag{4.1}$$

5. Randomly assign an aircraft type to every generated entrance time (since FCFS scheduling is assumed up to the TMA entry point). Only weight classes 1 to 5 will be allowed to use the runway.

Defining the entry times this way yields a user-defined average runway throughput over the total approach case duration while still allowing for a natural variation of time intervals between two consecutive aircraft entries.

4.4. OPTIMIZATION STRATEGY

Due to the significant problem type differences between the aircraft scheduling problem (ASP) and the Aircraft Trajectory Optimization Problem (ATOP), it is impossible to solve them as one and optimize both the ASP and the ATOP using a single optimization method. As was explained in section 2.4.3 it is advisable to determine an adequate strategy of optimization as it makes more sense to solve both problems in a sequential fashion, rather than simultaneously. This means that one has to determine in which order what part of the ASP and the ATOP has to be solved in order to find the best overall solution when both problems are combined. This section will explain the chosen optimization strategy.

4.4.1. DYNAMIC MODEL

Firstly, it is important to define how the approach problem is implemented as a dynamic model. As discussed in section 2.3.5, to allow for real-time optimization it is recommended to confine and scale down the problem in order to speed up computation times. For this thesis project it was chosen to make use of the scheduling horizon concept.

The benefit of using the scheduling horizon concept is that it enables the possibility of treating the dynamic approach problem as a static problem that can be recalculated whenever a new aircraft enters the scheduling horizon and thus new information to the optimization problem becomes accessible. As can be seen in figure 4.1 below the freeze horizon (FH) will be defined as Schiphol's TMA since all final approach trajectories must be known before entering the TMA. The scheduling horizon (SH) is the area closely around the TMA, where the information on exact TMA entry times of all arriving aircraft is assumed to be available. As explained in section 2.3.5, the scheduling horizon is where the aircraft sequencing and trajectory optimization process takes place. Whenever a new aircraft enters the SH all aircraft that are still within the SH will be resequenced. As an aircraft enters the FH its trajectory and thus its place in the landing queue will be "frozen". This means that its approach trajectory cannot be changed any longer and the aircraft is taken out of the scheduling process in the SH.

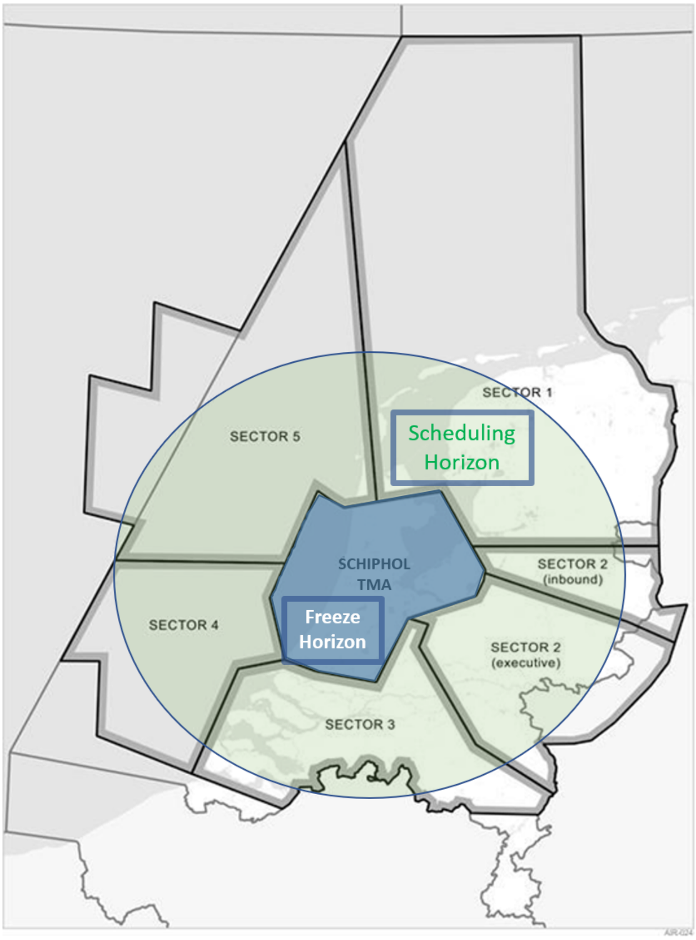


Figure 4.1: Placement of scheduling and freeze horizon

The exact size of the scheduling horizon will be determined in section 4.8.1 as this is a trade-off between computation time and scheduling optimality which is highly dependent on the way the ASP is solved.

4.4.2. STATIC MODEL

As explained in previous section, within the scheduling horizon the ASP and the ATOP can be solved as if it was a static problem with a fixed pool of aircraft of which the exact entry times are known in advance. This section will elaborate on the actual strategy that is chosen to solve both the ASP and the ATOP.

Section 2.4.3 already showed that the approach and the order in which the ASP and the ATOP are solved drastically affect the resulting performance indicators (i.e. fuel consumption, delay, runway throughput etc.). The research done by Sàma et al. [7] tested different approaches solving the ASP either before or after solving the ATOP (see table 2.4). The ASP was always solved for minimum individual delay while the ATOP was solved for either minimum fuel consumption or minimum travel time. It was concluded that solving the minimum-fuel ATOP first would yield the least average fuel consumption. However, solving the ASP first would result in significantly decreased delays. In the latter case it made no difference for which objective the ATOP was solved since travel times were dictated by the results of the ASP.

Although these findings are very useful when setting up an optimization strategy it is even more important to identify where the approach proposed by Sàma lacks significant optimization potential. Note that it is also important to keep in mind that although a static approach case is considered, certain strategy decisions can have an adverse effect when placed in a dynamic optimization environment, as was proposed in previous section. The following major drawbacks of Sàma's approach were identified:

- The objective of the ATOP was either to minimize travel time or to minimize maximum delay. However there are many more solutions possible that would offer a trade-off between fuel consumption and travel time. For every travel time between the minimum and maximum travel time a minimum fuel trajectory can be computed. It would be highly beneficial to use this information for solving the ASP.
- Conflicts were avoided when solving the ASP instead of actually solving potential conflicts in the ATOP.
- The ATOP and ASP are decoupled in a sense that trajectories are optimized disregarding other aircraft in the vicinity. Apart from the lost optimization potential this is not a feasible approach for this thesis research since the goal is to apply sequences (i.e. by overtaking) while flying a fixed horizontal approach path.
- The ASP was solved only for minimum delay. While for a static approach case this makes sense, in the long run (e.g. a dynamic approach case where new arrival aircraft are continuously added) this could actually lead to delay propagation. In this case solving the ASP for maximum runway throughput would make more sense.
- No fuel minimization objective was tested for the ASP. However, at times of low traffic levels where delay is not much of a problem, optimizing purely for minimum fuel consumption could be a choice to consider (at least for a short period of time using dynamic objectives).

In order to maximize the optimization potential when solving both the ASP and the ATOP and solving all of the identified drawbacks, for this thesis research a 3-module approach is proposed as briefly explained below. Detailed explanations of these three modules can be found at the end of this chapter in sections 4.5, 4.6 and 4.7 respectively.

1. **Module-1: Independent Aircraft Trajectory Optimization.** The first module will generate a database with fuel-optimal trajectories, between TMA entry and the runway threshold, for all achievable travel times. This will be done for each individual aircraft type assuming no other aircraft are in the vicinity (thus conflicts are neglected). For each aircraft type this will result in a "fuel curve" (see figure 5.2) that shows exactly what the minimum fuel consumption is for for each travel time between the earliest TOA and the latest TOA, assuming the aircraft can fly its own independent fuel-optimal trajectory. Note that these fuel curves are constant and therefore need to be computed just once.
2. **Module-2: Aircraft Scheduling Problem (ASP).** Using the fuel curves from Module-1 the ASP can now be solved for minimum delay, but also for minimum fuel consumption or a trade-off between the two. The ASP optimization objective will be adapted to the demanded runway throughput. In this module each aircraft within the scheduling horizon will be assigned with a target time of arrival (TTA) at the runway threshold resulting in an optimum landing sequence with respect to the chosen optimization objective.
3. **Module-3: Conflict Detection & Resolution.** Module-3 will solve the ATOP. Each aircraft must be assigned an optimal conflict-free approach trajectory that matches their respective TTA that resulted from module-2.
   Since both the entry time and the TTA is known, the corresponding independent fuel-optimal trajectory can be picked out of the database generated in module-1. These trajectories are first checked for

conflicts with aircraft whose trajectories were already frozen as they entered the TMA, which is the freeze horizon. In case of a conflict a fuel-optimal resolution by velocity and altitude changes will be computed.

This 3-module approach solves the static approach problem where the considered aircraft pool is defined by the number of aircraft that are currently in the scheduling horizon. This static approach problem is recalculated whenever a new aircraft enters the scheduling horizon. When an aircraft leaves the scheduling horizon and enters the TMA the corresponding trajectory solution and its place in the landing queue will be fixed and the aircraft is removed from the considered aircraft pool. This way a dynamic optimization environment is established. A flow chart of this static 3-module optimization strategy within a dynamic optimization environment can be found in figure 4.2.

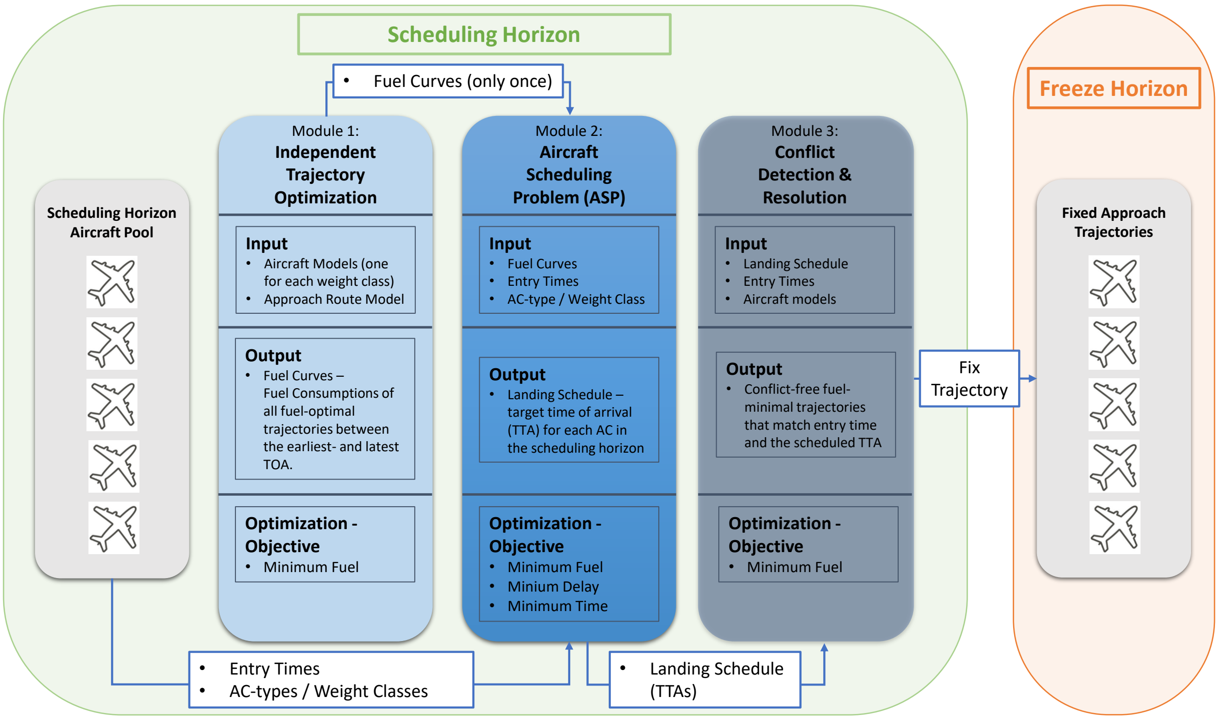


Figure 4.2: Flow Chart of static 3-module optimization strategy within a dynamic environment

A major difference with respect to previous optimization approaches is that the ATOP is divided in two parts, such that the ASP can use and benefit from fuel-optimal trajectory information but at the same time the ASP can still dictate the target times of arrival and thus the landing sequence of incoming aircraft. This optimization strategy therefore increases the optimization potential of the aircraft approach problem and enables the user to define a trade-off between fuel consumption and delay objectives based on pre-defined conditions, such as the demanded throughput.

4.5. MODULE-1: INDEPENDENT AIRCRAFT TRAJECTORY OPTIMIZATION

As was discussed in previous section, the first module of the designed optimization tool will generate a fuel curve for every considered aircraft type which contains valuable information, such as the time interval during which the aircraft's landing can be scheduled and the exact fuel consumption of the fuel-optimized approach trajectories for every target time of arrival (TTA) within this time interval. This section will give a detailed explanation on how these fuel curves are computed.

4.5.1. REQUIRED INPUT

To compute a fuel curve for the approach route of every considered aircraft the following information is required:

- **Approach Route Model.** The exact geometry of the approach route is required. This means that knowledge of the initial and final altitude, the total track length and also speed and altitude constraints at all points along the approach route have to be known. This information is defined in section 4.5.2.
- **Aircraft Models.** For every considered aircraft type an aircraft model must be available that accurately represents the aircraft's performance and flight characteristics to compute the corresponding fuel-optimal approach trajectory of that particular aircraft. The used aircraft models are presented in section 4.5.4.
- **Required Constraints on Aircraft Controls.** Apart from constraints imposed by the approach route model that mostly affect the aircraft model's states (altitude, airspeed etc.), several constraints must be placed on the aircraft model's controls in order to comply with certain approach procedures (e.g. continuous descent operations) and to make sure a computed trajectory is realistically achievable. These control constraints are defined in section 4.5.3.

All of the required input information listed above is constant during the entire approach case. For this reason, the fuel curves resulting from this module need to be computed only once, before the actual approach case optimization is started.

4.5.2. APPROACH ROUTE

When modelling an approach route and corresponding approach procedures upon which all scheduling and trajectory optimizations will be based, it is important to remember what the objective of this thesis research is. The goal is to sequence arrival aircraft in the last phases of descent inside the TMA and to compute corresponding fuel-optimal trajectories applying continuous descent operations (CDOs) within the limits of existing STAR designs for a single-runway system. Traffic sequencing will be achieved by small speed interventions and altitude changes.

Considering the research objective the "ARTIP 3B Transition" (one of Schiphol's night approach routes) is chosen as the reference approach route for this thesis research as it was designed for continuous descent operations featuring a single runway system. The night instrument approach map of this route can be found in appendices A.1 and A.2. It is assumed that all arrival traffic streams are merged before entering the TMA, such that all aircraft have to follow the same fixed lateral approach path from the initial approach fix (IAF) down to the final approach point (FAP) where the runway glide slope is intercepted. The geometry of the approach route is presented in table 4.1 below, where for each path segment the corresponding altitude and airspeed constraints are listed.

Path Segment	(IAF)	EH601	EH602	EH603	NIRSI	EH607	EH608	(FAP)
Track Length [NM]	28.4	6.2	6.1	6.3	4.9	4.0	4.6	n.a.
Max Airspeed [KIAS]	250	250	250	250	220	220	220	n.a.
Min Altitude [FL]	FL100	FL70	FL70	FL70	FL55	FL34	FL20	FL20
Max Altitude [FL]	FL100	FL100	FL100	FL100	FL55	FL55	FL34	FL20

Table 4.1: Path segments and corresponding altitude and airspeed constraints of the ARTIP 3B Transition

With this reference geometry in mind an approach route model must be set up that is tailored specifically to the research objectives of this thesis project. An important objective with respect to the approach route design is that advanced continuous descent operations will be applied where all aircraft follow the same lateral approach path. For this reason, it was chosen to solve the ATOP as a 2DT trajectory optimization problem (i.e. a 2-dimensional problem with time control), which only considers the track distance and altitude and neglects the aircraft's lateral position.
Further, it was chosen to use fixed initial and final airspeeds for all aircraft to prevent unresolvable conflicts between different aircraft types during conflict resolution in module-3. This will be explained in more detail in section 4.7.
Lastly, as can be seen from the geometry data in table 4.1 and the approach map in appendices A.1 and A.2, the altitude is constrained at just three positions during the approach route, namely at the IAF, the FAP and at the NIRSI-waypoint where the approach route turns inland.

Due to these factors the approach route is decided to be modelled as a straight horizontal approach track featuring two path segments enclosed by three waypoints. The geometry data of the approach route model is given in table 4.2 by means of trajectory position constraints and airspeed constraints.

Path Segment	IAF	Path Segment 1	NIRSI-waypoint	Path Segment 2	FAP
Track Position [NM]	0.0	0 - 47.0	47.0	47.0 - 60.5	60.5
Min Airspeed [KIAS]	250	180	180	180	180
Max Airspeed [KIAS]	250	250	220	220	180
Min Altitude [FL]	FL100	FL55	FL55	FL20	FL20
Max Altitude [FL]	FL100	FL100	FL55	FL55	FL20

Table 4.2: Approach route model geometry and airspeed constraints

4.5.3. Optimization Method & Setup

Since module-1 only considers the individual part of the ATOP and thus neglects potential conflicts, module-1 deals with a typical optimal control problem. Optimal control problems deal with finding a control law for a dynamic system that minimize a chosen optimization criterion over a period of time. This section will briefly discuss the chosen optimization method to solve this optimal control problem and will define in detail how the control problem is set up.

To solve the independent ATOP, a general-purpose optimal control software named "Gauss Pseudospectral Optimization Software" (GPOPS) will be used which uses a class of methods that is referred to as hp-adaptive Gaussian quadrature collocation. GPOPS is a MATLAB-based software and is a very suitable tool for solving non-sequential multiple-phase optimal control problems.

STATE VARIABLES

The independent ATOP is implemented as a 2-phase optimal control problem, where the two phases solve the control problem for the two path segments of the approach route model's geometry that was defined in section 4.5.2. As discussed, the ATOP is considered a 2DT trajectory optimization problem, which means that only track distance and altitude are considered while the lateral track position can be neglected. Combined with the aircraft velocity and aircraft weight, these four state variables form the optimal control problem for the ATOP. The limits of these four state variables, given in table 4.3 below, represent the geometry of the approach route model. Note that the airspeed state variable corresponds to the aircraft's true airspeed while the maximum airspeeds on the approach track are expressed as calibrated airspeeds. Therefore the airspeed state limits are dependent on the aircraft's current altitude. The aircraft's initial approach weight W_{app}, at the start of phase 1, depends on the type of aircraft. W_{app} is a nominal value for each type of aircraft and is defined in equation 4.7 of section 4.5.4

Phase / path segment		**Phase 1**			**Phase 2**		
		initial	bounds	terminal	initial	bounds	terminal
Track Distance s [m]	min	0	0	87044	87044	87044	112046
	max	0	87044	87044	87044	112046	112046
Altitude h [m]	min	3048	1676	1676	1676	609.6	609.6
	max	3048	3048	1676	1676	1676	609.6
True Airspeed V [m/s]	min	148.51 (250KCAS)	free	free	free	free	95.30 (180KCAS)
	max	148.51 (250KCAS)	... (250KCAS)	122.48 (220KCAS)	122.48 (220KCAS)	... (220KCAS)	95.30 (180KCAS)
Weight W [kg]	min	W_{app}	free	free	free	free	free
	max	W_{app}	W_{app}	W_{app}	W_{app}	W_{app}	W_{app}

Table 4.3: State variables and respective state limits of the ATOP optimal control problem

CONTROL VARIABLES

In order to control these four state variables and keep the system (i.e. the aircraft) between the established limits, the control variables must be defined. Since a 2DT trajectory problem is considered, the aircraft's bank angle and yaw angle can be neglected and are assumed to equal zero at all times. For this reason, the aircraft can be assumed to be controlled using just two control variables, the engine thrust and the flightpath angle which is controlled through the aircraft's elevator angle that results directly from the chosen aircraft model, as will be explained in section 4.5.4. The two control variables and their respective limits are given in table 4.4 below.

Phase / path segment		**Phase 1**			**Phase 2**		
		initial	bounds	terminal	initial	bounds	terminal
Engine Thrust setting η [normalized (0-1)]	min	0	0	0	0	0	0
	max	1	1	1	1	1	1
Flightpath Angle γ [deg]	min	-25	-25	-25	-25	-25	-25
	max	25	25	25	25	25	25

Table 4.4: Control variables and respective control limits of the ATOP optimal control problem

EQUATIONS OF MOTION

Next, the influence of the control variables on the state variables must be defined through the equations of motion, which are essentially the state variables derivatives. The equations of motion are not only a function of the state - and control variables but are also dependent on the aircraft's engine- and performance parameters that follow from the chosen aircraft model, that is discussed in section 4.5.4. In this case these parameters are thrust T, drag D and fuel flow ff. The equations of motion are defined in equations 4.2-4.5.

$$\dot{s} = V \cdot cos(\gamma) \tag{4.2}$$

$$\dot{h} = V \cdot sin(\gamma) \tag{4.3}$$

$$\dot{V} = g_0 \cdot \frac{(((T \cdot n_{eng} - D) \cdot \frac{V}{W}) - \dot{h})}{V} \tag{4.4}$$

$$\dot{W} = -ff \cdot n_{eng} \cdot g_0 \tag{4.5}$$

PATH-CONSTRAINTS

Further, path constraints must be defined to ensure the trajectory optimization method computes an optimal trajectory that also meets the existing approach procedure and is also flyable regarding passenger comfort. This last point becomes very clear when comparing found trajectories with and without applying necessary path constraints in figure 5.1 which will be further explained in the results chapter. Since CDA procedures apply for this approach case, the goal is to find optimum trajectories with continuously descending vertical profiles with minimal level flight segments that are executed at the lowest possible thrust settings. To achieve this two path constraints were defined in table 4.5 that prevent the aircraft from both accelerating and climbing. Applying these path constraints not only ensures a minimum thrust continuous descent but also prevents an aircraft from performing a so called "cyclic cruise", as will be explained in section 5.1

Phase / path segment		**Phase 1**	**Phase 2**
Acceleration $\dot{V}$ $[m/s^2]$	min	-9.81	9.81
	max	0	0
Vertical Speed $\dot{h}$ $[ft/min]$	min	-2500	-2500
	max	0	0

Table 4.5: Path constraints applied on the ATOP optimal control problem

COST-FUNCTION

Finally, the cost-function must defined in order to optimize the trajectory for minimum fuel. Since the burned fuel is the only factor affecting the aircraft's weight, the fuel consumption can simply be determined by comparing the initial- and final value of the weight state variable. Therefore, the Mayer cost component can be defined as:

$$J = W_{initial} - W_{final} \tag{4.6}$$

4.5.4. AIRCRAFT MODEL

In order to solve the equations of motion that were defined in equations 4.2-4.5, a model that accurately simulates an aircraft's flight performance characteristics is required . For this task the BADA (Base of Aircraft Data) aircraft performance model developed by Eurocontrol will be used. Eurocontrol developed BADA especially to support modelling and simulation tools for the purpose of ATM research and development. BADA provides theoretical model specifications and related specific datasets to accurately stimulate the behaviour of any type of aircraft [40].

Since the number of available Matlab-based BADA aircraft type models is limited for this thesis project, only one aircraft type model is considered per weight class. It is assumed that other aircraft types within the same weight class feature similar flight performance characteristics. The used aircraft type model for each weight class can be found in table 4.6.

Weight Class		**BADA Aircraft Type Model**	**Initial Approach Weight W_{app} [N]**
Super Heavy	"A"	Airbus A380-861	3,634,400
Upper Heavy	"B"	Airbus A340-313	1,707,700
Lower Heavy	"C"	Boeing B763	1,200,500
Upper Medium	"D"	Boeing B738	588,440
Lower Medium	"E"	Fokker F100-650	341,030

Table 4.6: BADA aircraft type model used for per aircraft weight class

BADA models contain fixed nominal weights (e.g. MTOW, MLW, MZFW, OEW and FC) for each different aircraft type. These weights can be used to estimate the weight at which an aircraft will initiate its final approach. It is assumed that this initial approach weight W_{app} is identical for each approaching aircraft of the same aircraft type. It is estimated that an aircraft initiates its final descent with 75% of its maximum payload and with 10% of its fuel capacity left, as given in equation 4.7. The estimated W_{app} of the chosen aircraft type models can be found in table 4.6.

$$W_{app} = (OEM + 0.75 \cdot (MTOW - OEW - FC) + 0.1 \cdot FC) \cdot g_0 \tag{4.7}$$

Lastly, since this is an approach problem, all aircraft are assumed to be in approach configuration. The corresponding BADA flap models of the respective aircraft types were used. Since the optimal control optimization method does not allow to change between different configurations during the optimization process, it is assumed that the flap settings are fixed over the entire approach route.

4.5.5. ASSUMPTIONS & SIMPLIFICATIONS

Before diving into details on how the independent approach trajectories are optimized for minimum fuel resulting in a fuel curve for each considered aircraft type, the most important assumptions and simplifications that were made are listed below:

- **Approach Route and Trajectory**
  - The scope is limited to the final approach phase of the flight, which starts at the IAF and ends at the FAP.
  - Since advanced CDA procedures will be followed (i.e. the horizontal approach path is fixed), the trajectory optimization will be 2-dimensional and thus only consider track length and altitude. Therefore the horizontal approach path is assumed to be straight.
  - Fixed entry speeds and final speeds at the runway threshold are assumed.
  - All traffic streams are merged before entering the TMA which means that within the TMA every arriving aircraft will follow the same horizontal approach path down to the runway threshold.

  - ACs are assumed to be able to perform TBO with 100 percent accuracy (meaning ACs arrive at waypoints at the exact time they were scheduled to be there).
  - Minimum Fuel Consumption is the only considered optimization objective during the computation of optimal independent trajectories.

- **Aircraft Model (BADA)**
  - Nominal initial aircraft weight is considered for each approach.
  - Weather effects, such as windspeeds, are neglected.
  - All aircraft are assumed to be in an approach configuration over the entire approach route. Corresponding flap models of the specific aircraft type were used.
  - Only one aircraft type model is considered per weight class. It is assumed that other aircraft types within the same weight class feature similar flight performance characteristics.
  - It is assumed that only five different weight classes, ranging from "Super Heavy" to "Lower Medium", will are allowed to use the runway. "Light" aircraft are not considered in this approach problem.

4.5.6. FUEL CURVE COMPUTATION APPROACH

Finally the fuel curves can be computed. As discussed, a fuel curve shows the minimum fuel consumption to fly the approach route for the entire range of possible travel times. These fuel curves are aircraft type specific. This section will explain the used approach to compute the fuel curves for the five chosen aircraft type models. A flow chart of this approach can be found in figure 4.3.

1. **Travel Time Interval.** Firstly the interval of possible travel times must be determined in which the aircraft is capable to cover the approach route. To do so the optimal control problem described in section 4.5.3 will be solved twice, once with the minimum time optimality criterion and once with the maximum time optimality criterion. The two resulting solutions describe the earliest and latest possible arrival time at the FAP. From this the travel time interval is determined since the tested aircraft model will be capable of reaching the FAP at any time in between these two landing times.

2. **Fuel-optimal Trajectories.** Now that interval of possible travel times is determined, the optimal control problem can be solved for the minimum fuel criterion. The problem will be iteratively solved with fixed final times that cover the entire travel time interval in steps of one second. This results in a discrete fuel curve of the respective aircraft type model. An example of this can be found in figure 5.2 in the results chapter. 5-second intervals are shown here for visualization purposes.

3. **Polynomial Regression.** Although the fuel curve is now computed, it still cannot be used for the aircraft sequencing optimization in module-2. The reason for this is that the fuel curve is now a discrete curve that is divided in 1-second intervals. As will be explained in section 4.6.3 the aircraft sequencing process will use a SQP-based local solver to find optimal landing sequences. This means that the fuel curve must be differentiable twice in order to be used by this local solver. For this reason the discrete fuel curve was fitted with a 20th-degree polynomial to make the fuel curve continuous over the entire interval travel times. An example fuel curve of the Airbus A380 aircraft model can be found in figure 5.2, showing both the discrete and the polynomial fitted curve.

4. **Normalized Fuel Curves.** Lastly, an option to normalize the fuel curves with respect to an aircraft's fuel-optimal landing time was added. This option can be used to analyse how sequencing results are affected when removing weight class prioritization from the sequencing problem and thus maintaining a sense of fairness between different aircraft weight classes. These normalized fuel curves are plotted in figure 5.5.

Note that a fuel curve is aircraft type specific. This means that the described approach must be repeated for each of the five chosen aircraft type models.

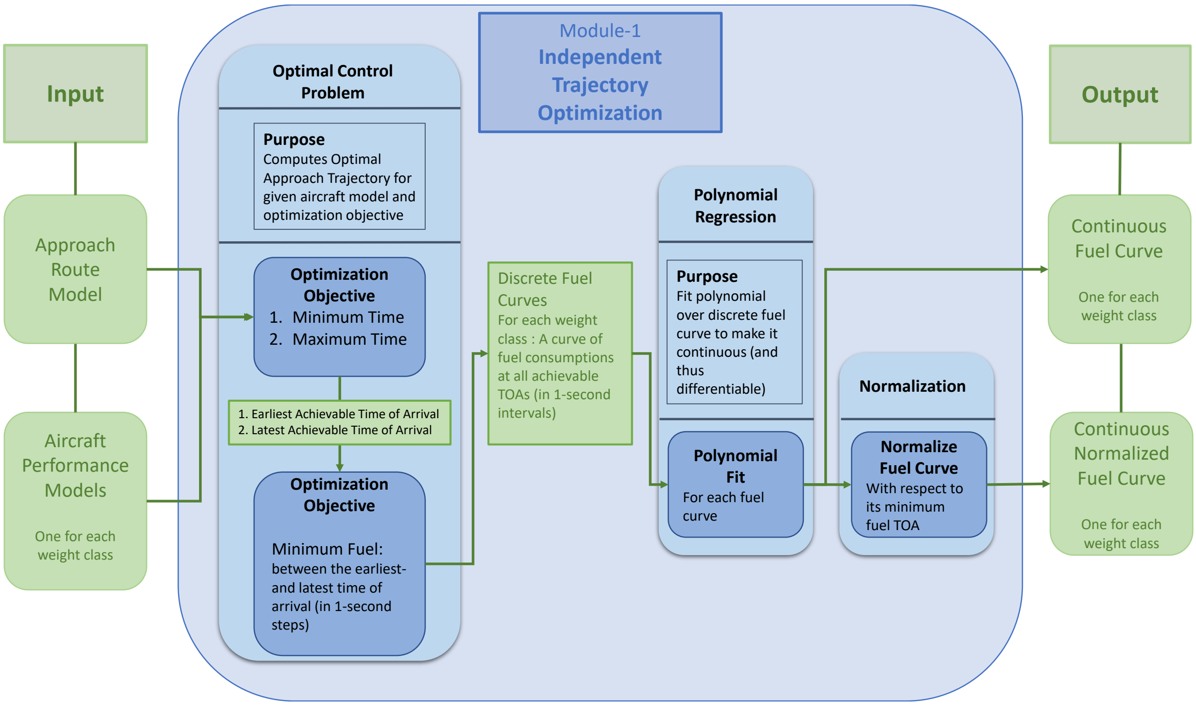


Figure 4.3: Flow Chart of Independent Trajectory Optimization Approach

4.6. MODULE-2: SOLVING THE AIRCRAFT SCHEDULING PROBLEM (ASP)

As discussed in section 4.4.2, this module will solve the aircraft scheduling problem (ASP) only considering the aircraft that are currently within the scheduling horizon (SH). Therefore the ASP is treated as a static scheduling problem with a fixed pool of arriving aircraft. Every aircraft will be assigned with a target time of arrival (TTA) resulting in a landing sequence that is optimal with respect to the chosen optimality criterion. This section will give a detailed explanation on how the landing schedule is optimized for these different optimality criteria.

4.6.1. REQUIRED INPUT

The required input for module-2, in order to compute optimal landing schedules is the following:

- **Aircraft Types** Firstly, the aircraft type of each aircraft within the SH must be known since the corresponding weight class dictates exactly what wake separation distances are required with respect to leading and following aircraft. Wake separation is by far the most significant factor in aircraft scheduling.

- **Fuel Curves.** Corresponding to each aircraft, the fuel curve of that aircraft type must be available. The fuel curves are not only necessary when optimizing the landing schedule for minimum fuel but also for post-processing minimum delay and minimum time scheduling solutions, since within this solution it is likely that some TTAs can be moved such that less fuel is consumed without affecting the delay or time objectives.

- **TMA Entry Times.** Lastly, for each aircraft within the SH, the expected time of entering the TMA (i.e. arriving at the IAF) must be known. This information is required in order to move the corresponding fuel curves to the correct positions in time.

4.6.2. ASP - OPTIMIZATION METHOD

Next, an optimization method must be chosen to solve the ASP for a chosen optimization objective. However, before a decision is made it is important to realize what kind of an optimization problem the ASP actually is. Generally, the ASP can be defined as a combination of two sub-problems:

1. **Landing Sequence.** The first sub-problem is finding the optimal landing sequence in which aircraft arrive at the runway threshold. The landing sequence mainly defines the required wake separation distances between approaching aircraft and therefore dictates the bounds between which each individual aircraft is allowed to land. For this reason, this sequencing sub-problem is of a combinatorial kind and features discrete constraints (the wake separation distances). Therefore this problem can best be solved using a heuristic optimization method (i.e. a global solver) as was discussed in section 2.4.2.

2. **Landing Schedule / Landing Times.** With the landing sequence solution and the found bounds in between which individual aircraft are allowed to land, the actual landing schedule (i.e. the exact landing times of each individual aircraft) is still undefined. The landing schedule directly affects the objective-function of the ASP, as moving a landing time two seconds back will increase delay by an equal amount of time and change the fuel consumption by an amount that can be computed by use of the fuel curves. For this reason, this second sub-problem that solves the actual landing schedule should be solved by calculus-based optimization methods (i.e. a local solver) .

These two sub-problems are closely interrelated since it is impossible to know whether a landing sequence is optimal without computing the optimal landing times within this sequence. At the same time, knowing the optimal landing times within one tested sequence, it is still unknown whether this solution is optimal without comparing the solution to the optimal landing schedule of all other possible landing sequences. However, since both sub-problems are optimization problems of a very different kind, it will be challenging to find the global minimum to an optimization objective using just one single optimization method:

- **Problem with local solvers:** A local solver will only find a minimum within the sequence imposed by the initial guess (local minimum). It will never try to change the sequence itself since in order to do so it has to pass a less optimal (or in this case even infeasible) region first.

- **Problem with global solvers:** These methods randomly select values for the design vector to test. After a finite number of these randomly chosen initial guesses the solution with the best objective value is considered the answer. Unfortunately, there is no way to know how many tests must be run in order to find the definite best solution to the problem (i.e. the global optimum). Also, since the initial guesses are randomly chosen, in case of the combinatorial nature of this ASP it is unavoidable that a certain landing sequence is tested more than once, leading to the same local minimum multiple times and thus a significant performance penalty. To conclude, using solely global solvers to find the solution to the ASP can become very computationally intensive while it is not certain that the found solution is actually the global minimum.

For these reasons, the only viable method to solve the ASP seems to be a hybrid-optimization method, combining global and local solvers. As stated by Betts, to find the global minimum for problems with discrete variables, such as this highly combinatorial sequencing problem, the only way to decide if a candidate solution $\hat{\mathbf{x}}$ is the answer is by comparison with all other possible candidate solutions [35].

For the ASP, this means that all possible aircraft sequences must be tested for optimality. In order to do so, MATLAB's SQP-algorithm will be used to compute the optimal landing schedule within a certain landing sequence. Each possible landing sequence will be tested for optimality by using them as initial guess to the SQP-algorithm. In short, this means that the SQP-algorithm will solve the scheduling problem based on multiple initial guesses, defined by the possible landing sequences. The exact approach to this optimization method will be explained in section 4.6.3.

4.6.3. ASP OPTIMIZATION APPROACH

This section will explain exactly how the ASP is solved for a fixed pool of aircraft. The optimization approach will be explained in consequential steps. A flow chart of the ASP optimization approach can be found in figure 4.5.

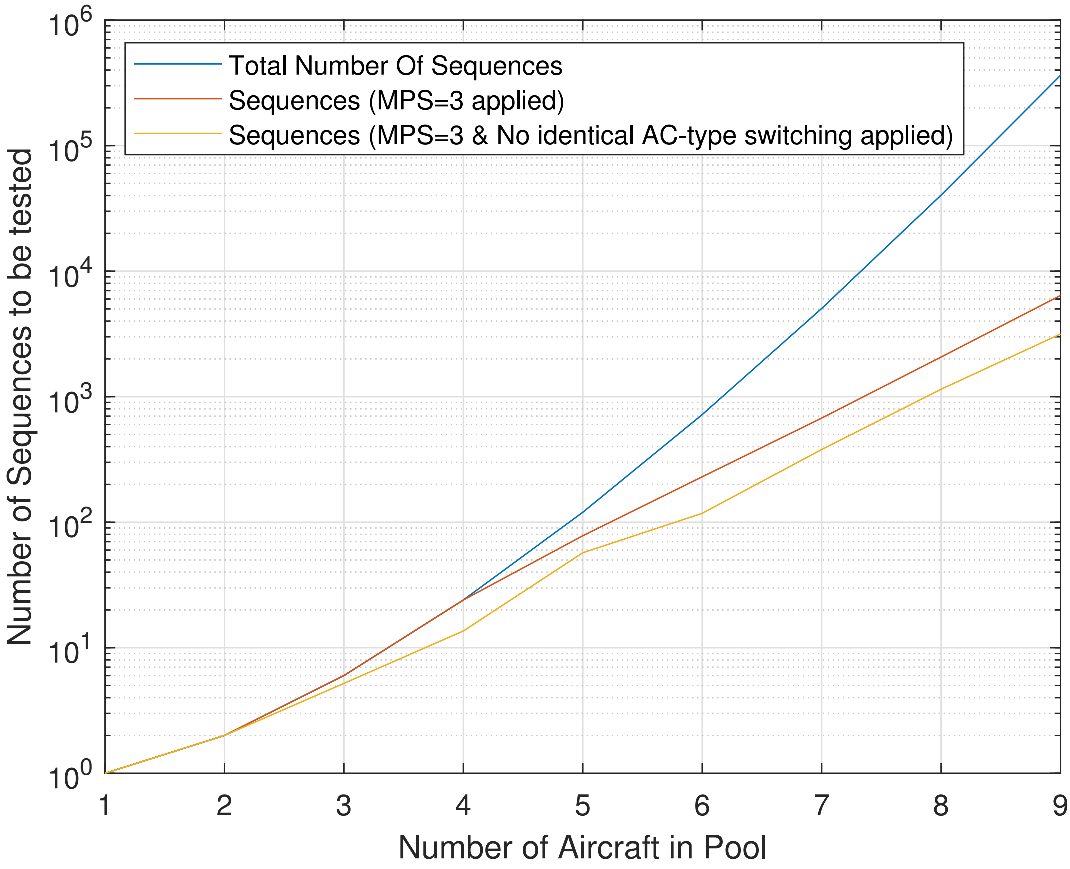


Figure 4.4: Number of landing sequences to be tested for optimality as function of aircraft pool size

1. **Determine Potentially Optimal Sequences.** Firstly, all possible landing sequences must be determined for the fixed pool of aircraft. For example a 7-aircraft pool contains 7!=5040 different landing sequences. Fortunately, some of these sequences can be considered non-optimal right away and do not have to be tested.

   - **Identical AC-types never switch positions.** Aircraft of the same type, when scheduled right after each other in the landing queue, will always land in the order of their TMA entry since there is no possibility that switching places would yield a more optimal overall sequence, whether the objective is fuel-, delay- or capacity-based. To ensure that this sequencing rule is valid, it was tested 100 times for all of the three optimization objectives. The resulting optimum sequence was never between the sequences that were discarded by this rule.
   - **Constrained Position Shifting (CPS).** If found necessary one can apply constrained position shifting (CPS, explained in section 2.3.3) to reduce the number of sequences that will be tested for optimality. All sequences in which a landing position shift exceeds the maximum position shift (MPS) are discarded.

Figure 4.4 shows that these two sequencing rules can significantly reduce the number of sequences that will be tested for optimality. However, the most significant factor is the size of the considered aircraft pool. Section 4.8.1 explains how it can be beneficial to artificially limit the size of the considered aircraft pool.

2. **Transform Sequences to Initial Guesses (Start Points).** Next, each of the remaining landing sequences must be transformed to an initial guess that can be used by the SQP-algorithm to test the corresponding sequence for optimality. This initial guess must be an array of landing times that represent their corresponding landing sequence, lie within the optimization boundaries (i.e. earliest and latest achievable landing times) and meet the optimization constraints (i.e. time-based wake separation). The following approach is applied to transform a landing sequence to an array of landing times:

   - **Place landing times as early as possible.** In order of landing sequence (starting with the aircraft that will land first), place the landing time as close to the leading aircraft's landing time as possible, such that time difference is exactly the wake separation distance.
   - **Check if landing times are within bounds.** If this landing time is before the aircraft's earliest achievable landing time (i.e. its lower bound), move the landing time back to this lower bound.
   - **Remove infeasible landing sequences.** Since the landing time of a leading aircraft affects the possible landing time of the trailing aircraft, it is very possible that for this trailing aircraft the earliest possible landing time due to wake separation is after the latest achievable landing time. In this case the corresponding landing sequence is not achievable and will be discarded.

   Using this approach, landing sequences that are not achievable are removed before testing them for optimality. The remaining sequences are transformed to an initial guess that represents the minimum time solution of the respective landing sequence, since all aircraft are moved to their earliest possible landing times within this sequence.

3. **Optimization.** Finally, all remaining initial guesses (corresponding each to one unique achievable landing sequence) can be optimized for the chosen optimization objective using the SQP-algorithm.

4. **Solution.** The solution with the best objective value is considered as the answer. In case the optimization objective is *minimum delay* or *minimum time*, it is possible that multiple solutions with the same (minimal) objective value remain. In this case a post-optimization will be performed (see step 5).

5. **Post-Optimization.** In case multiple solutions still remain (for example 4 solutions that yield no delay at all) the solutions will be optimized once more for a *minimum fuel* objective, adding a constraint that fixes the previously found minimum objective value. Again, the solution with the best objective value is considered as the solution. An example, comparing the scheduling solution with and without a post-optimization for minimum fuel can be found in figures 5.8 and 5.9 respectively.

Following this approach, one can say with certainty that the globally optimal solution to the ASP is found every time since all possible landing sequences are tested for optimality. However, it has to be noted that the solution considers only the currently available information, the current aircraft pool. An optimal solution of the static ASP problem does not guarantee that it will be optimal in a dynamic case as well.

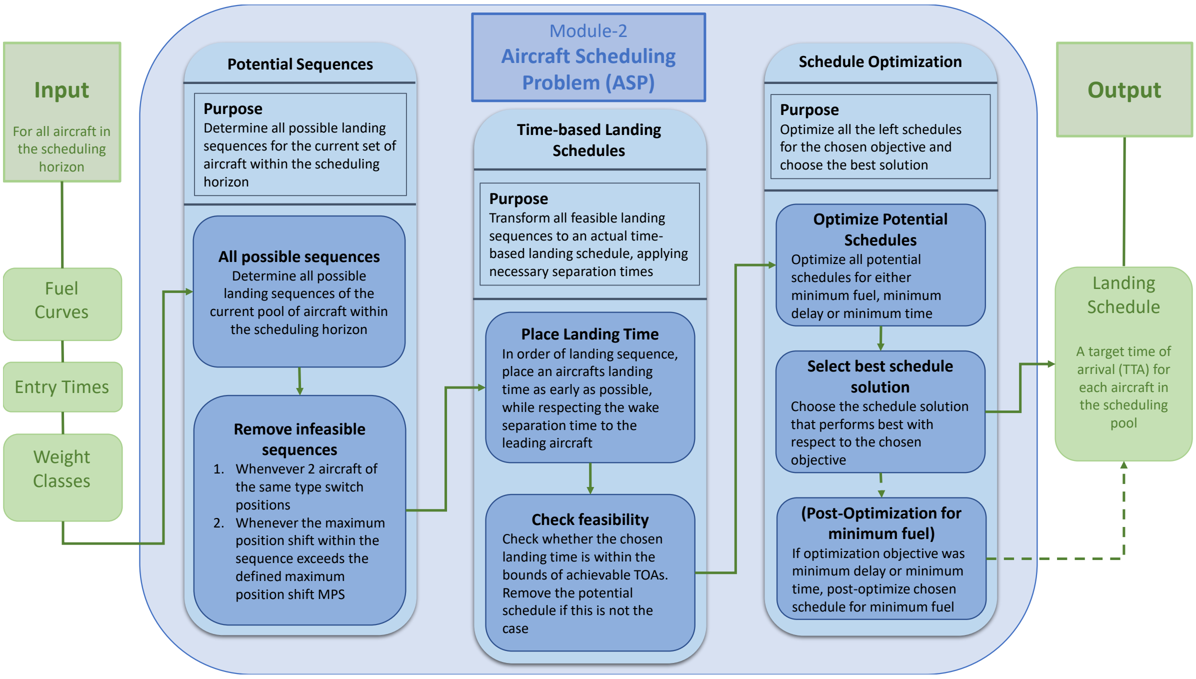


Figure 4.5: Flow Chart of ASP Optimization Approach

4.6.4. ASSUMPTIONS & SIMPLIFICATIONS

- Time-based wake separation distances are applied via sequencing constraints for reasons explained in section 2.3.2.
- TMA entry times are fixed and thus cannot be affected.
- It is assumed that only five different weight classes, ranging from "Super Heavy" to "Lower Medium", will are allowed to use the runway. "Light" aircraft are not considered in this approach problem.
- Only one aircraft type model is considered per weight class. It is assumed that other aircraft types within the same weight class feature similar flight performance characteristics.

4.7. MODULE-3: CONFLICT DETECTION & RESOLUTION

This last module that is part of solving the static approach problem deals with conflict detection and conflict resolution. The optimal landing schedule that was computed in module-2 must now be applied, meaning that a conflict-free and fuel-optimal trajectory, between the entry time at the IAF and the scheduled arrival time at the FAF, must be assigned to each of the aircraft in the considered arrival pool. In some cases the fuel-optimal independent trajectories found in module-1 can be used, however in many cases these trajectories will be in conflict with other aircraft trajectories in the vicinity and therefore must be adapted to resolve the existing conflict. This section will give a detailed explanation on how conflicting trajectories are detected and how they are resolved.

Remember that this module is part of solving the static approach case considering only the aircraft that are currently in the scheduling horizon. However, since a dynamic approach case is considered, only the trajectory solution of the aircraft that is currently entering the freeze horizon will be fixed during this optimization run. All remaining aircraft are still open for resequencing in case a new arriving aircraft would required a different scheduling solution.

4.7.1. REQUIRED INPUT

In order to compute a conflict-free fuel-optimal approach trajectory, meeting the landing time computed in module-2, for each aircraft of which the approach trajectory will be frozen during this optimization run, the following information is required as input:

- **Landing Schedule.** Firstly, the landing schedule that was computed by module-2 is required. The scheduled landing times, together with the entry times of the considered aircraft pool, determine the time interval of the fuel-optimal conflict-free approach trajectory that must be computed by this module.
- **Independent Fuel-Optimal Trajectories.** Based on this time interval, the corresponding independent fuel-optimal trajectory, that was computed by module-1, will be used as an initial guess. In case this trajectory does not cause any conflicts it can be kept the same and does not require re-computation.
- **Fixed Trajectory Data of Aircraft in Freeze Horizon.** Lastly, it is required to know which aircraft trajectories are currently active in the freeze horizon after being fixed during previous optimization runs. These trajectories are required to check new trajectories for conflicts and to use as separation constraints in case occuring conflicts must be resolved.

4.7.2. METHOD FOR FUEL-OPTIMAL CONFLICT RESOLUTION

As briefly mentioned in section 4.4.2, this module will solve the actual aircraft trajectory optimization problem (ATOP) which is no different from the independent trajectory optimization problem solved by module-1, except that trajectories must now account for other aircraft in the vicinity as well. For this reason the same 2-phase optimal control problem as defined in section 4.5.3 will be solved, again using the GPOPS optimal control solver. However, this time a constraint is added that restricts trajectories to fly within minimum radar separation distance ($3NM$) of another aircraft.

In case of a detected conflict, by applying this constraint to the optimal control problem defined in section 4.5.3, GPOPS will try to resolve this conflict with an adapted approach trajectory aiming to stay as fuel-optimal as possible. How this separation constraint function is exactly formulated will be explained in section 4.7.5.

4.7.3. NON-COOPERATIVE CONFLICT RESOLUTION

Whenever an aircraft enters the freeze horizon (FH), its trajectory will be fixed. Therefore, by definition, the conflict resolution between an aircraft in the FH and an aircraft that is not yet assigned with a trajectory is non-cooperative. However, before an aircraft's trajectory is fixed it would make sense to check whether this trajectory will create conflicts with other aircraft in the scheduling horizon, whose trajectories are not yet fixed either. This way a more cooperative resolution can be found that is more fuel-optimal.

Unfortunately, a cooperative way of resolving conflicts leads to compatibility issues with the GPOPS optimal control solver. GPOPS solves optimal control problems within the boundaries of defined phases. These boundaries are defined by the state variable limits at both, the initial- and final time of a phase. An aircraft's trajectory is represented by four state variables. Thus for a cooperative conflict resolution between two aircraft, the four states of both aircraft must be optimized, which requires a 8-state-variable optimal control problem. The issue however is that the initial- and final time of a phase are dictated by the FH entry time and landing time of just one particular aircraft. However, since the two aircraft do not have identical entry times and landing times, there will be a time mismatch between the state variables of both aircraft. Therefore, it was chosen to resolve conflicts non-cooperatively by computing and freezing trajectories in order of the arriving aircraft's landing times.

4.7.4. CONFLICT RESOLUTION APPROACH

This section will explain how conflicts are detected and resolved (i.e how the ATOP will be solved). Note that in this approach fuel-optimal conflict-free trajectories will only be computed for aircraft of which trajectories must be assigned and fixed during this optimization run. It makes no sense to do this for all aircraft within the scheduling horizon since their trajectories must remain open for change for future optimization runs.

1. **Find Active Trajectories within Freeze Horizon (FH).** Determine which of the trajectories that were frozen during earlier optimization runs are currently active. In other words, which aircraft are currently in the FH and have not yet landed? This information is required from step 6(a) and onwards.

2. **Determine which aircraft are added to Freeze Horizon.** It is not required to fix an aircraft's approach trajectory until it enters the FH. However in some cases an aircraft that enters the FH is scheduled to land after one (or more) aircraft that is still in the scheduling horizon (SH) and thus will be overtaken within the FH. In this case it was chosen to freeze both, the trajectory of the aircraft entering the FH and all aircraft within the SH that are scheduled to land at an earlier time. The number of aircraft that need to be assigned with a fixed trajectory during this optimization run is therefore variable.

3. **Select Independent Fuel-Optimal Trajectory based on Travel Time.** For each aircraft for which a conflict-free fuel-optimal trajectory must be computed during this optimization run, select an independent fuel-optimal trajectory computed by module-1 that meets the aircraft's entry time and landing time. This trajectory will serve as an initial guess during conflict resolution (or will serve as final solution in case no conflicts were detected).

4. **Interpolate Added Trajectories.** Trajectories must be made continuous in order to be checked for conflicts at each point in time. A "Piecewise Cubic Hermite Interpolating Polynomial" (PCHIP) is used to interpolate between the nodes of the selected independent fuel-optimal trajectory solution of each aircraft.

5. **Determine Freezing Order.** Since conflicts are resolved non-cooperatively an order has to be determined in which trajectories will be made conflict-free and frozen. It seems most obvious to compute and freeze trajectories in order of entrance times into the freeze horizon. However, doing so could lead to scenarios where a trajectory must be found in which the final descent the the FAF is in between two

other fixed trajectories. Since this often leads to unresolvable conflicts, it was chosen to compute and freeze trajectories in order of landing times.

6. **(Loop: for each approach trajectory to be fixed) Detect & Resolve Conflicts.** In order of trajectory freezing (i.e. in order of landing times), for each aircraft of which the trajectory will be frozen during this optimization run, repeat the following steps:

   (a) Compare the PCHIP-interpolated independent fuel-optimal trajectory (from step 4) to the currently active and frozen approach trajectories (from step 1) and check whether a conflict exists (i.e. whether, at the same time both, the aircraft's horizontal and vertical separation distance is violated at any time between the aircraft's TMA-entry time and landing time).

   (b) If no conflict was detected freeze (fix) the trajectory. Else, if a conflict was detected with one or more existing frozen trajectories, proceed to the next step.

   (c) Resolve the detected conflicts by running the constrained optimal control problem in GPOPS, optimizing the trajectory for minimum fuel with the added minimum radar separation (MRS) constraints.

   (d) Freeze the computed conflict-free fuel-optimal approach trajectory by interpolating between its nodes using PCHIP and adding the trajectory to the currently active freeze horizon trajectories.

A flow chart explaining this approach can be found in figure 4.6.

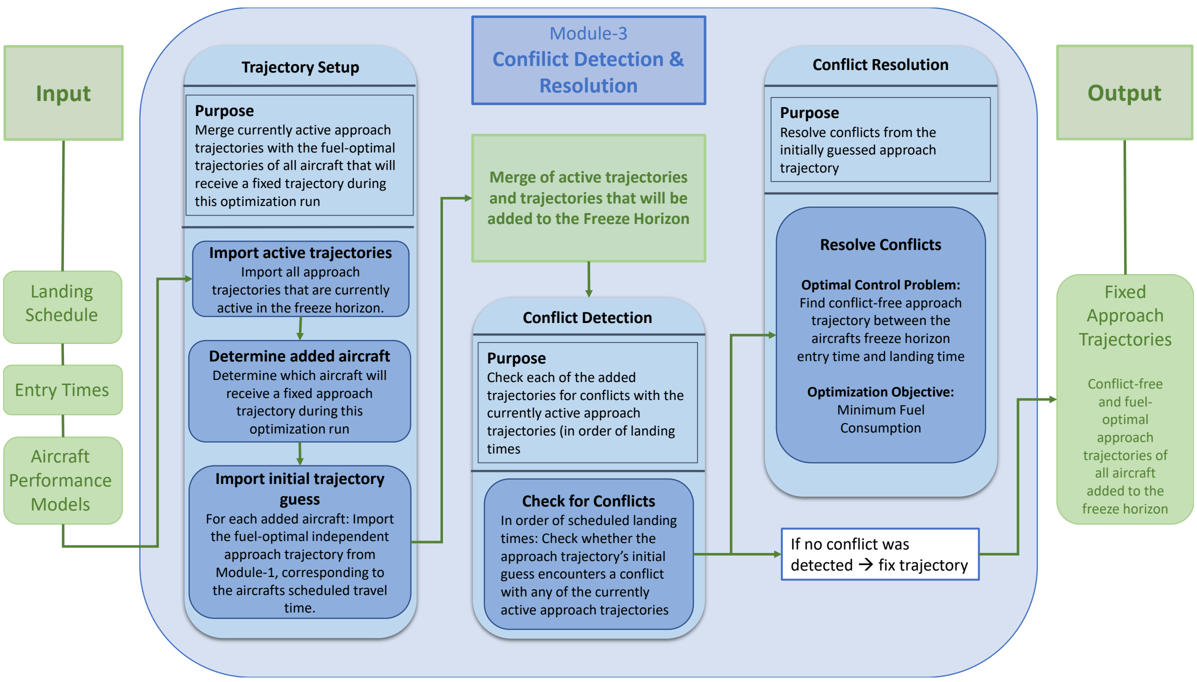


Figure 4.6: Flow Chart of Conflict Detection & Resolution Approach

4.7.5. FORMULATION OF THE SEPARATION CONSTRAINT FUNCTION

This section will explain how the separation constraint function is formulated in order to work within the optimal control problem used in module-1. The constraint function will be distance based and will only consider minimum radar separation (MRS) to other aircraft in the freeze horizon. Since aircraft follow different vertical approach paths and often need to pass each other in the vertical plane it makes no sense to consider time-based wake separation as well until the FAP, from where on all aircraft will follow an exactly identical approach path over the ILS glideslope. Since the wake separation at the FAP is already established by the applied landing schedule, wake separation is not considered during the trajectory optimization.

As stated in section 2.2.1, two aircraft are in conflict with each other when both of the following conditions apply at the same time:

- The horizontal distance between both aircraft is less than 3NM.
- The vertical distance between both aircraft is less then 1000ft.

This means that the definition of a conflict if of a logical kind, since it requires some sort of AND-statement. Unfortunately, local solvers such as GPOPS cannot handle logical operators since these operators make the respective constraint function discontinuous. Discontinuous functions cannot be differentiated twice over the entire interval which means the solver will not be able to find the required function gradient at all places.

Apart from this, a gradient-based solver such as GPOPS requires more information than only whether the trajectory is in conflict or not. The solver also requires gradient information of the constraint function. In other words, GPOPS needs to know how severe the conflict is and in what direction the trajectory must change in order to resolve this conflict.

4.7.6. REMOVAL OF SEPARATION PATH CONSTRAINT DISCONTINUITY

For the reasons explained above, the separation constraint function must be made continuous by combining both, the horizontal- and vertical separation into one constraint function that is differentiable twice over the entire optimization interval. This is done to have only one constraint function instead of two with a non-continuous AND-statement that GPOPS cannot handle, since a constraint is either active or inactive depending on a systems state and control variables. It cannot be made dependant on other constraints.

In order to achieve this it was chosen to normalise both, the horizontal- and vertical distances between aircraft with respect to the minimum required horizontal and vertical radar separation. Using these normalized distances, the separation path constraint function can be formulated as just one single normalized separation distance:

$$D = \sqrt{ds^2_{normalized} + dh^2_{normalized}} \tag{4.8}$$

Where the normalized separation distance D must be larger or equal to one:

$$D >= 1 \tag{4.9}$$

This formulation defines an ellipsoid-shaped conflict volume around an aircraft, as shown in figure 4.7. Although in reality the conflict volume around an aircraft should be disk-shaped, as explained in section 2.2.1, this simplification was necessary for GPOPS to be able to handle the separation path constraint. As can be noticed in figure 4.7, the defined ellipsoidal conflict volume does not contain the entire MRS-volume, especially when the conflicting aircraft are flying at different altitudes. Although increasing the size of the defined ellipsoid could solve this problem, this solution would drastically increase the horizontally required separation when aircraft are flying at the same altitude and thus is not a viable solution.

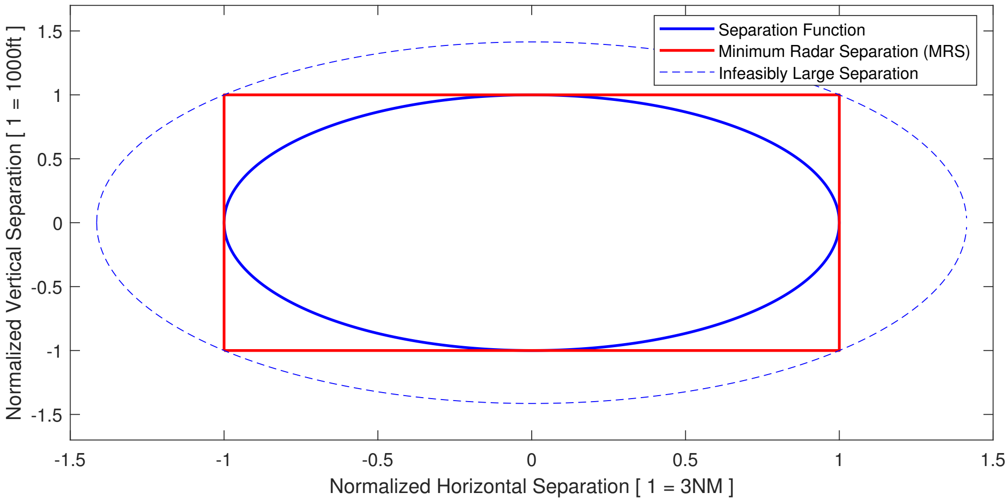


Figure 4.7: Normalized Conflict Volume around an aircraft: actual required separation (red) and continuous simplified separation function (blue)

4.7.7. HANDLING UNRESOLVABLE TRAJECTORY CONFLICTS

In some cases trajectory conflicts simply cannot be solved within the defined state variable boundaries and constraint limits. Especially since it was chosen to resolve trajectory conflicts non-cooperatively, in some cases it is possible that no conflict free trajectory can be computed due to previously made trajectory decisions that are already fixed. For this reason a decision was made to not consider the conflict volume around an aircraft as a path-constraint but instead use a barrier function to add separation violations as a penalty to the cost function described in equation 4.6.

The penalty function P is defined as:

$$P = A \cdot e^{(-3.465 \cdot D)} \tag{4.10}$$

Where D is the normalized separation distance as defined in equation 4.8 and A is a coefficient that depends on the aircraft type and is used to scale the penalty function with respect to the aircraft types fuel consumption. The graphical representation of the penalty function is plotted in figure 4.8. As can be seen in the figure the penalty is negligibly small until the normalized separation approaches 1. As soon as the conflict volume is violated (D<1) the penalty significantly increases. This penalty function aims at finding a trajectory that conflict free, however when no conflict-free trajectory is possible provides the possibility to allow separation violations while keeping their severity to a minimum.

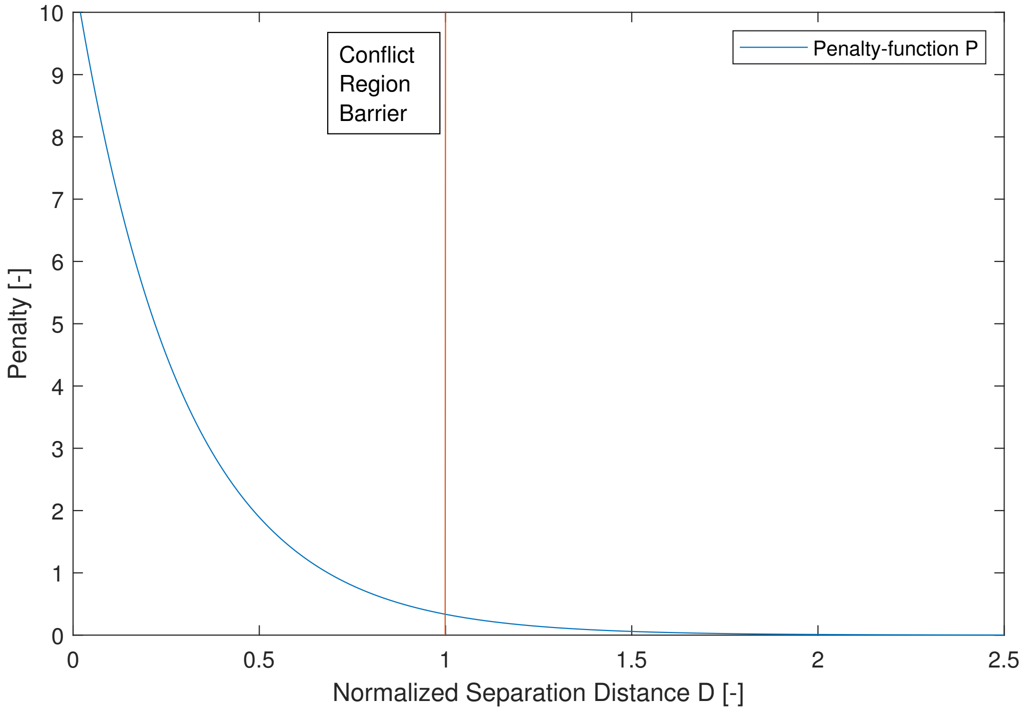


Figure 4.8: Separation penalty $P(A = 1)$ as barrier function of the normalized separation distance D

Implementing the conflict constraints through the objective function has two major benefits with respect to implementing them through path constraints.

- **Less restrictive.** Allowing small constraint violations, suboptimal trajectory solutions due to path constraints are prevented. In rare cases an implementation through path constraints cannot find a conflict free trajectory at all.
- **Reduced computation time.** Due to the restrictive nature of a path constraint, in quite a few cases computation times are excessively large since no fuel-optimal trajectory can be found that meets all constraints. A less restrictive implementation of the conflict constraint through the objective function also significantly reduces these computation times

Note that despite the benefits to this implementation, it must be checked whether it does affect the objective function too much, such that fuel-optimality becomes too less of a priority. Multiple tests have shown that the added separation penalty in equation 4.10 is never more than 5% of the total objective function such that the fuel-optimality of the computed trajectory solutions is hardly affected. Apart from this it will be tested whether adding a small safety factor to the conflict zone (e.g. increasing the normalized minimum separation distance D from $D = 1.0$ to for example $D = 1.1$) will yield fuel-optimal trajectory solutions without violating the actual conflict zone of $D = 1.0$. This case-study can be found in section 5.3.1 in the results chapter.

4.7.8. ASSUMPTIONS & SIMPLIFICATIONS

This section lists the assumptions and simplifications that were made in module-3 in order to detect and resolve trajectory conflicts.

- **Method & Approach**
  - Non-cooperative conflict resolution is assumed for method compatibility reasons, as explained in section 4.7.2
  - A "Piecewise Cubic Hermite Interpolating Polynomial" (PCHIP) was used to interpolate between nodes of found optimal trajectory solutions.
  - Minimum radar separation is assumed during conflict resolution. Since aircraft do not follow the same vertical descent profile, time-based wake-separation is only considered and applied at the FAF, since from here on the ILS glideslope is intercepted from where aircraft will follow an exactly identical 3D approach trajectory.
- **Conflict Separation Constraint**
  - For path-constraint continuity reasons, the conflict volume around an aircraft is assumed to be of an ellipsoidal shape. In reality, this volume should be disk-shaped.
  - The minimum radar separation (MRS) can be violated if necessary, however this constraint violation must be kept as small as possible.

4.8. DESIGN OF A DYNAMIC OPTIMIZATION ENVIRONMENT

So far the implementation would be able to solve a static approach problem. This means: fixed pool of aircraft (fixed number of ACs and fixed weight classes) and fixed TMA entrance times that are all known in advance. In real life however, this information is not known in advance, as ACs of random weight classes arrive with their exact TMA entrance times known only minutes before arriving there. Therefore the optimization tool must be adaptable to a variable approach situation, recomputing landing sequences and approach trajectories whenever new information is available, as was explained in section 4.4.1.

4.8.1. FREEZE HORIZON APPLICATION

As discussed in section 4.4.1, the freeze horizon concept will be used, in combination with constraint position shifting (CPS), to confine the problem to a size that can be solved in real-time. Generally it is aimed to keep the scheduling horizon as small as possible in order to keep computation times down to a minimum. At the same time it cannot be too small as this could significantly affect the optimality of the scheduling solutions.

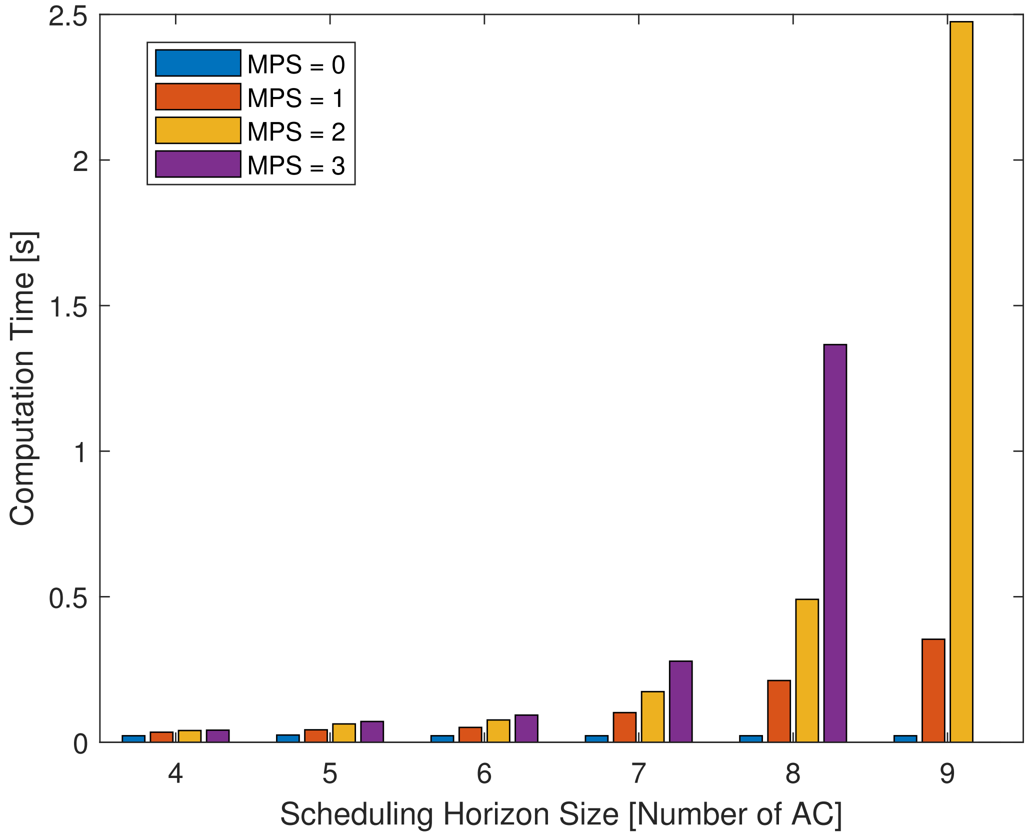


Figure 4.9: ASP Computation Time for various scheduling horizon aircraft pool sizes and maximum position shifts (MPS) on an Intel Core i5-6300U processor

SCHEDULING HORIZON SIZE

Typically in a freeze horizon concept, the size of both the scheduling horizon and the freeze horizon are defined by their geometric boundaries. For this research however it was chosen to define the size of the scheduling horizon by the number of aircraft it entails. The reason for this is that the number of aircraft within a scheduling horizon with geometric boundaries is highly dependent on the traffic density. Since the optimization tool must be able to handle any amount of required throughput, it makes much more sense to consider a fixed number of aircraft for scheduling at a time. The question that must be answered is what number of aircraft would provide acceptable computation times without affecting the optimality conditions of the scheduling solutions. Since the considered aircraft pool size only affects the optimality of the scheduling solutions, isolated tests of module-2 will be run in order to determine an adequate scheduling horizon size.

The average computation time of a single static approach case for various aircraft pool sizes (i.e. scheduling horizon sizes) combined with various maximum position shifts for CPS is given in figure 4.9. Up to an aircraft pool size of 8 aircraft the computation times are very manageable, considering the time between subsequent aircraft entries into the TMA are between one and two minutes. However, for aircraft pool sizes of 9 and up the used laptop system ran out of memory. For this reason a maximum aircraft pool size of 8 aircraft will be considered for the remainder of this report.

It becomes much harder to qualitatively compare the optimality of corresponding scheduling solutions. To compare minimum-fuel-based scheduling solutions, test cases must be run at a constant and high value of required throughput, since at lower throughput levels no difference to a FCFS scheduling solution will be visible. Unfortunately this cannot be tested since the maximum throughput levels that can be reached is very different when using different scheduling horizon sizes.

Therefore, it was tested what the maximum achievable average throughput level is that can be scheduled for various scheduling horizon sizes. A throughput level is considered achievable if the optimization tool is able to schedule at least 200 aircraft without the need for aircraft holding for each of 5 consecutive test runs. The results, which can be found in figure 5.12 in the results chapter, point out that a scheduling horizon size of 8 aircraft is optimal with a maximum position shift of 3 places. This decision will be explained in section 5.2.3.

4.8.2. DYNAMIC OPTIMIZATION OBJECTIVES

Lastly, since the optimization tool will work in a dynamic aircraft approach environment, the required throughput will slightly vary over the time of the test case. Take for example the test performed in figure 5.12 of previous section. If an average throughput of 48 aircraft per hour is chosen for the 200 aircraft test case, this means all of the 200 aircraft will enter the TMA within a $200AC \cdot \frac{48AC}{60min} = 160min$ interval. However, since the actual entry times are generated randomly, as explained in section 4.3, this does not mean that a new aircraft entry occurs exactly once every 75 seconds. The actual required throughput can vary within this 160 minute interval.

For this reason, it makes sense to not use the same optimization objective for every optimization run during the entire aircraft approach case. Rather, one could try to dynamically tailor the optimization objective to the current scheduling horizon properties. In short this would mean to optimize the landing schedule for minimum fuel whenever the capacity allows for it, but switching to a maximum throughput scheduling objective as soon as, for example, delay starts to build up. Using a dynamic scheduling objective strategy like this will allow for fuel minimal scheduling of a large dynamic approach problem while still providing an option to overcome sudden traffic peaks that require a different scheduling objective.

The exact condition that determines when to switch from fuel-minimal aircraft scheduling to a more throughput-favouring scheduling objective must yet be determined. Two potential dynamic objective strategy options are listed below.

1. **Based on scheduling horizon delay.** As can be seen from the computed fuel curves in figure 5.5, there is very little room for delay to build up until the latest achievable TOA is exceeded. For this reason it makes sense to switch from fuel-minimal scheduling to time-minimal scheduling as soon as any delay is predicted in the scheduling horizon in order to quickly achieve more scheduling headroom and remove this delay. As soon as no more delay is predicted for the aircraft within the scheduling horizon, the minimum fuel scheduling objective can be applied again in the next optimization iteration.
2. **Based on required throughput.** Another strategy option could be to base the scheduling objective on the currently required throughput. Knowing the amount of aircraft in the scheduling horizon and their entry times, one could predict whether a fuel-minimal scheduling objective can achieve this required throughput. The actual objective switching threshold must be determined but will probably be based on the theoretical throughput limit for fuel-minimal scheduling.

Results with regards to which of these strategies works best can be found in section 5.2.6.

5

Results

In this chapter all relevant results of the designed optimization tool for arrival sequencing and trajectory optimization are presented. Firstly, sections 5.1, 5.2 and 5.3 will analyse and explain the results of the tool's individual modules respectively, to verify their proper working. Section 5.4 will then present the overall results when running the entire optimization tool to simultaneously solve the aircraft scheduling problem (ASP) and the aircraft trajectory optimization problem (ATOP) in a dynamic environment.

5.1. Independent Aircraft Trajectory Optimization

As discussed in section 4.5, Module-1 is supposed to generate a so called fuel curve for each individual aircraft type, that shows the exact fuel consumption at every achievable target time of arrival. This is done by computing a fuel-optimal approach trajectory between the earliest- and latest achievable time of arrival in steps of one second. This section will present the resulting fuel curves and analyse the corresponding fuel-optimal trajectories to ensure the proper working of Module-1.

5.1.1. Effect of applied path-constraints

Before analysing the fuel curves that resulted from Module-1 of the optimization tool, the proper working of the trajectory optimization must be verified. Therefore it must be checked whether the computed fuel-optimal trajectory solution meets the state limits and control limits defined in section 4.5.3 and whether the applied path-constraints on the airspeed, the acceleration and the vertical velocity actually yield a continuous descent approach (CDA) trajectory that is realistically flyable. To do so, the vertical approach profile of the computed fuel-optimal trajectory of the Airbus A380 aircraft model will be analysed, that is plotted in figure 5.1. The plot shows both, the vertical profile with and without the applied path constraints.

Without the applied path-constraints the resulting trajectory solution meets all state- and control limits of the defined optimal control problem but is far from realistically flyable and does not represent a continuously descending approach. The trajectory solution however does verify the correct working of the fuel-optimal trajectory optimization since the corresponding vertical profile shows the typical properties of a so called "cyclic-cruise" or "periodic cruise" which. Without constraints on the acceleration and vertical velocity of an aircraft, periodic cruise is the most fuel efficient way of flying. The reason for this is that thrusting takes place at a higher speeds since an optimal periodic cruise consists of a minimum rate of descent glide followed by a full thrust ascent at maximum rate of climb [41].

Applying the path constraints on the aircraft preventing it from both accelerating and climbing removes this periodic cruise pattern and yields a continuous descent approach profile as required. As expected when optimizing for minimum fuel, the aircraft stays at the highest altitude for as long as possible before a final continuous descent is started. The reason why there is a short second level flight is because the aircraft is required to enter the last part of the approach route (the second phase of the optimal control problem) at a fixed altitude as was defined in section 4.5.2

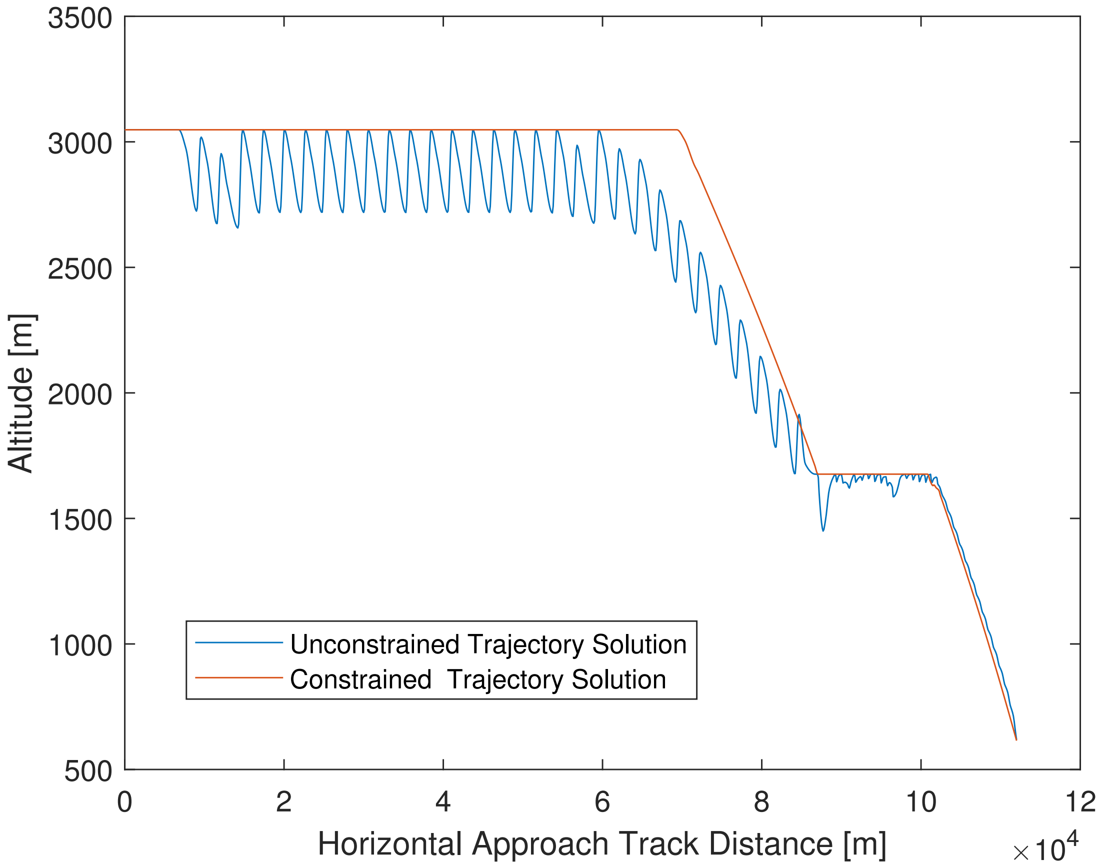


Figure 5.1: Vertical profile of the unconstrained- and constrained fuel-optimal approach trajectory solution of a Airbus A380 aircraft model

5.1.2. FUEL CURVES

Next, the fuel curves resulting from the independent trajectory optimization of Module-1 will be analysed. An example fuel curve of the Airbus A380 aircraft model can be found in figure 5.2. This figure shows both, the raw fuel data obtained from the discrete time steps at which a fuel-optimal trajectory was computed and the poly-fitted fuel curve corresponding to this data. As explained in section 4.5.6 this polynomial regression is necessary since the scheduling method used in Module-2 requires continuous fuel curve data.

The Airbus A380 example fuel curve shows 2 distinct areas, a parabolically decreasing part between the earliest achievable time of arrival af 801 seconds and the fuel-optimal arrival time at 1061 seconds followed by a steeply increasing almost linear part until the latest achievable time of arrival at 1111 seconds. These 2 areas can be explained by analysing the vertical profiles and their corresponding airspeeds for the computed trajectories of 5 different arrival times along the fuel curve. These vertical profiles and corresponding airspeeds are plotted in figures 5.3 and 5.4 respectively.

- **TTA: 801-1061 seconds:** Between the earliest achievable arrival time at 801 seconds and the fuel-optimal arrival time at 1061 seconds the vertical approach profile is almost identical and shows the typical behaviour of a fuel-optimal continuous descent approach. The aircraft stays at maximum altitude for as long as possible before descending down to the FAF. The only difference between the three trajectories is the airspeed at which the trajectory is performed which explains the slowly decreasing fuel consumption. The earliest arrival time is achieved by flying the approach trajectory at the maximum allowed airspeed. The trajectory corresponding to the 866 seconds arrival time is performed at a slightly lower airspeed during the first path segment while the airspeeds in the second segment are equal due to the decreased maximum airspeed in this last segment. Lastly, the fuel-optimal approach trajectory is performed at the lower and more economic maximum range airspeed. Note that the airspeed plot shows true airspeed (TAS), hence why the airspeed plot shows a decreasing airspeed whenever altitude is lost.

- **TTA: 1061-1111 seconds:** By delaying the arrival time beyond the fuel-optimal arrival time, it is not fuel efficient to decrease the airspeed even further. As can be seen in from the vertical profile, the arrival time is further delayed by losing altitude earlier during the approach. As expected the latest possible arrival time is achieved by descending to the minimum altitude (i.e. FL55 and FL20) as soon as possible while flying at the lowest possible airspeed. This early loss of altitude causes the steep increase in fuel consumption between the arrival times of 1061 and 1111 seconds.

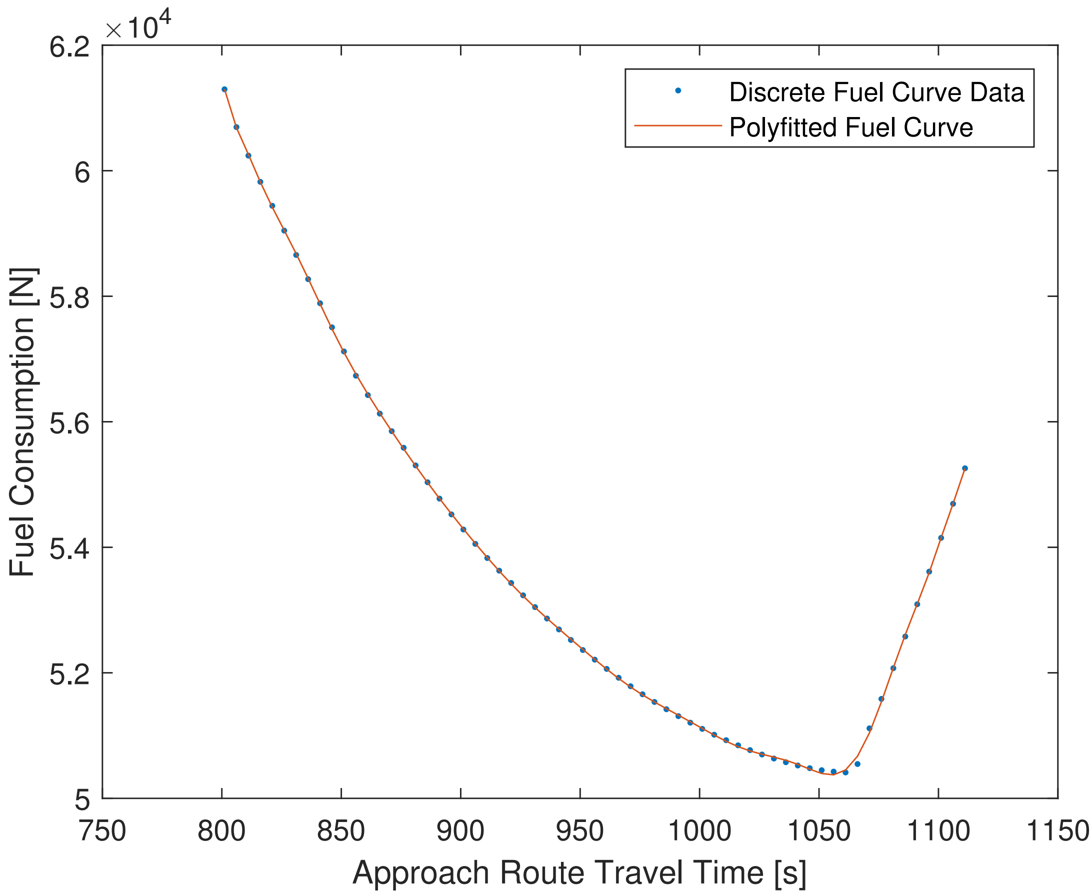


Figure 5.2: Fuel Curve Data of the Airbus A380 aircraft model (5-second intervals for visualization purposes)

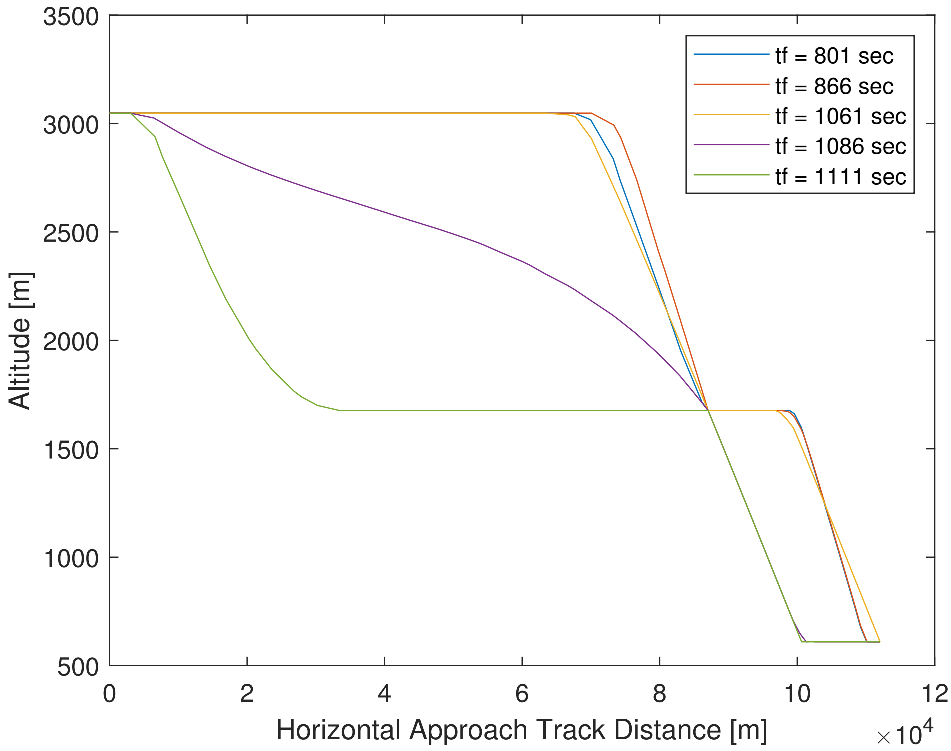


Figure 5.3: Vertical profiles of various approach route travel times (Airbus A380)

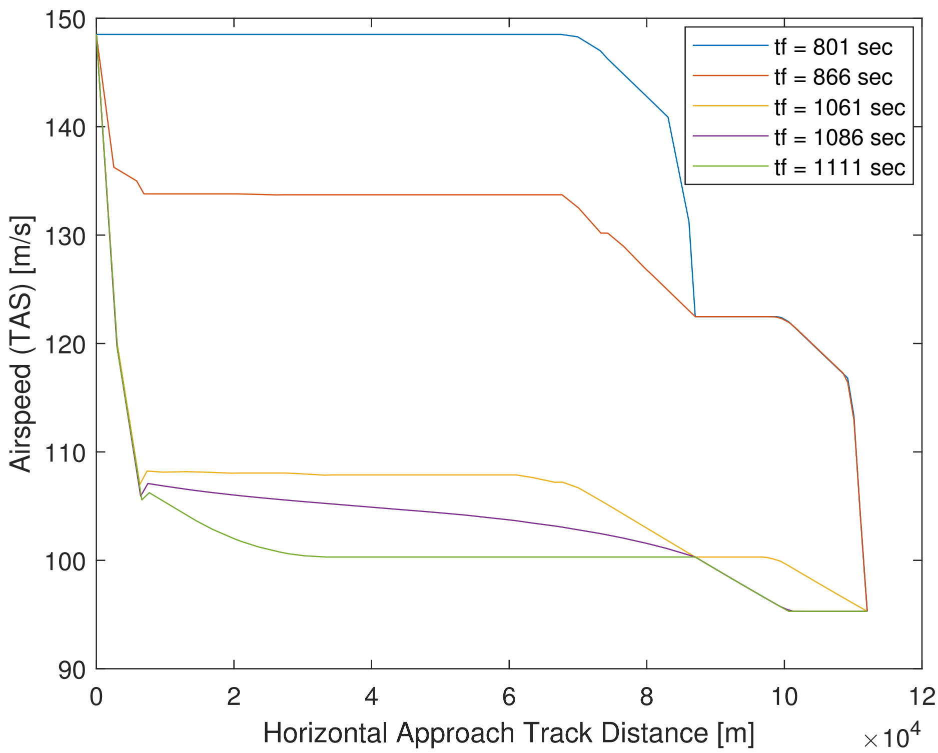


Figure 5.4: Airspeeds corresponding to the vertical profiles in figure 5.3 (Airbus A380)

5.1.3. Normalized Fuel Curves

As discussed in section 4.5.6 it makes sense to add an option to normalize the fuel curves in order to maintain a sense of fairness between different aircraft weight classes and analyse how this then affects the scheduling results. The normalized fuel curves for the reference aircraft of each of the 5 weight classes are plotted in figure 5.5 below.

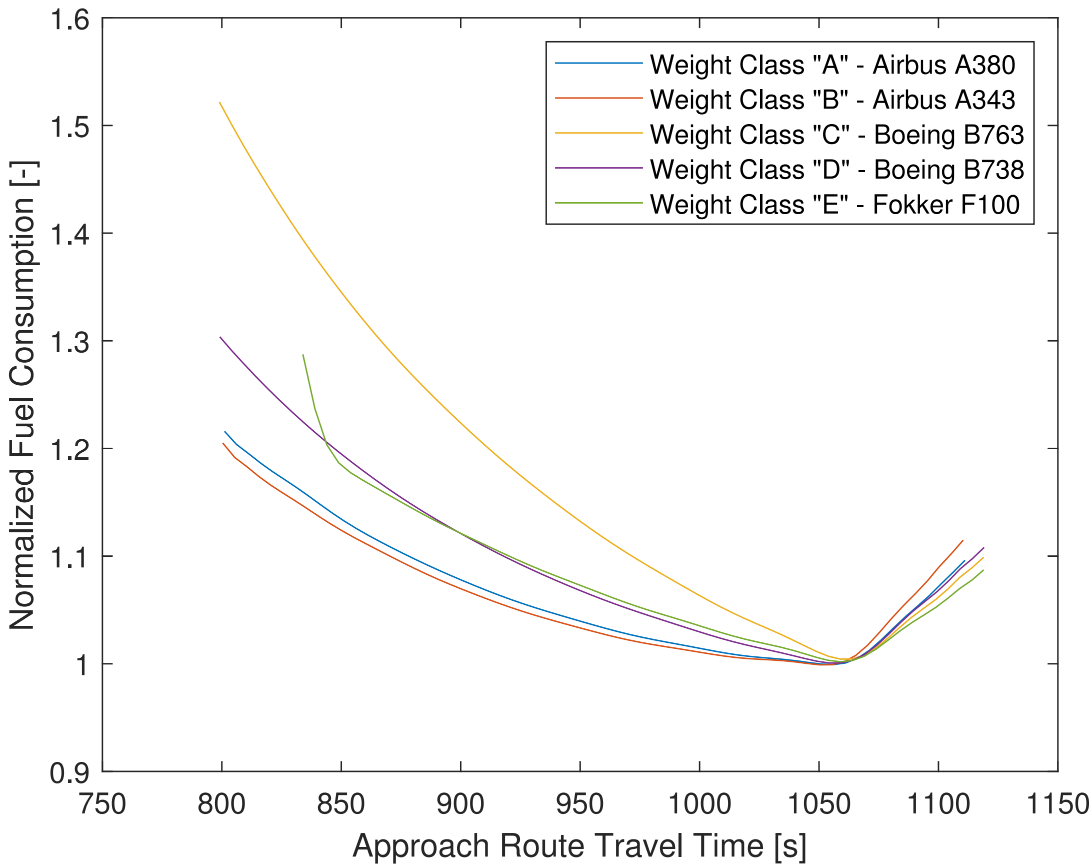


Figure 5.5: Normalized fuel curves for 5 different weight class aircraft models

The normalized fuel curves show a very similar behaviour although for the Boeing B763, performing a trajectory at higher, less economic airspeeds seems to have a more significant impact on the aircraft's fuel consumption in comparison with the other aircraft's fuel curves. The length of the achievable arrival time interval is almost equal. This was expected since initial- and final airspeed of the approach route are fixed and the airspeed is limited between the maximum allowed airspeed an the final airspeed which serves also as the minimum airspeed since no acceleration is allowed by means of the applied path constraints. The fuel-optimal travel time of the approach route is also very similar between all aircraft models because the minimum-fuel airspeed lies beneath the final approach airspeed. Since the model is not allowed to accelerate this causes the fuel-optimal approach trajectories of all aircraft models to have an almost identical vertical approach profile and airspeed profile. Lastly, the Fokker F100 aircraft model is not able to achieve an earliest time of arrival around the 800 second mark. Most likely the reason for this is that the aircraft is not able to reach the maximum allowed airspeed of 250KIAS while being in approach configuration with flaps extended. This would also explain the steep kink in the beginning of the F100 fuel curve, as the model is forced to enter at an airspeed of 250KIAS.

5.2. AIRCRAFT SEQUENCING

5.2.1. THE STATIC ASP - VERIFICATION OF SCHEDULING SOLUTIONS

Firstly, in order to verify whether the designed optimization algorithm in Module-2 to solve the ASP is working correctly, the scheduling solution of a static 8-aircraft approach case will be analysed and compared for the minimum-fuel, minimum-delay and minimum-time objectives. The approach case properties are given in table 5.1 below. The corresponding results can be found in figures 5.6 to 5.9 and table 5.2. These figures show the fuel curves of each individual aircraft in the scheduling horizon with their respective target time of arrival (TTA) that was computed to be optimal. Remember that, as discussed in section 4.6.3, the minimum-fuel objective is the only optimization objective that yields a unique optimal scheduling solution. The reason for this is that for both, the minimum-time and minimum-delay objectives, there often are more than one possible solution resulting in the same minimum latest time of arrival or minimum average delay. If for example multiple solutions exist with the same latest time of arrival, all of these solutions (i.e. sequences) will be post-optimized for minimum-fuel, while keeping the latest time of arrival fixed.

Aircraft Entry	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Freeze Horizon entry-time [sec]	41.19	80.99	122.20	165.40	205.49	249.66	290.14	335.65
Aircraft Weight Class	C	D	B	D	C	A	A	D
Earliest Time of Arrival [sec]	840.16	880.12	922.57	964.53	1004.45	1050.78	1092.25	1134.78
Latest Time of Arrival [sec]	1160.16	1200.12	1232.57	1284.53	1324.45	1360.78	1401.25	1454.78

Table 5.1: Static 8-aircraft approach case properties

Optimization Objective (minimum-fuel post-optimization)	**Minimum-Fuel** n.a.	**Minimum-Delay** enabled	**Minimum-Time** disabled	**Minimum-Time** enabled
Last aircraft arrival time [sec]	1455	1346	1300	1300
Total Fuel Consumption [kg]	25006	25736	27428	27091
Average Delay [sec]	8.125	0	0	0
Found optimal sequence [place in queue]	1 2 4 3 5 6 7 8	1 3 2 4 6 7 8 5	1 2 3 5 4 7 8 6	1 2 4 3 5 7 8 6

Table 5.2: Static 8-aircraft approach case results

- **Minimum-Fuel.**Starting off with the minimum fuel solution given in figure 5.6, as expected the aircraft of the highest weight classes are given priority and thus scheduled as close as possible to their minimum fuel consumption arrival times. For this reason the last entering aircraft (AC-entry 8), which is of the lowest weight class, was scheduled at its latest possible arrival time at 1455 seconds in order to make as much room as possible for the higher weight class aircraft. The downside of this is that aircraft 8 must therefore be delayed by 65 seconds.

- **Minimum-Delay.** The minimum average delay solution of the same approach case, given in figure 5.7, is not much different to the minimum fuel solution, except for the fact that the delay of aircraft 8 is removed. To do so aircraft 8 is scheduled to land fifth in the queue right before the higher weight class aircraft 5, 6 and 7, which were delayed as much as possible. Delaying them further would cause aircraft 8 to be delayed. The removal of the 65 second delay of aircraft 8 resulted in a 2.92% increase in overall fuel consumption but decreased the time to land all 8 aircraft by 109 seconds, with respect to the minimum-fuel scheduling solution.

- **Minimum-Time without fuel post-optimization.** The minimum-time scheduling solution is radically different with respect to the previously discussed solutions. For solely the minimum-time objective all that matters is that the last aircraft arrival is scheduled as early as possible. As explained earlier, it is possible that multiple solutions exist that yield the same minimal latest arrival time. Figure 5.8 shows one of these solutions. The scheduling solution managed to land all 8 aircraft 155 seconds sooner than the minimum fuel solution but at the cost of a 9.69% increase in overall fuel consumption. No delay is caused.

- **Minimum-Time with fuel post-optimization.** Lastly, if all minimum-time solutions are post optimized for minimum-fuel a unique scheduling solution is found where the last arriving aircraft still arrives at 1300 seconds but the corresponding overall fuel consumption is minimized. This minimum-time scheduling solution that was post-optimized for minimum fuel consumption can be found in figure 5.9. The increase of overall fuel consumption with respect to the minimum-fuel schedule is 8.34%.

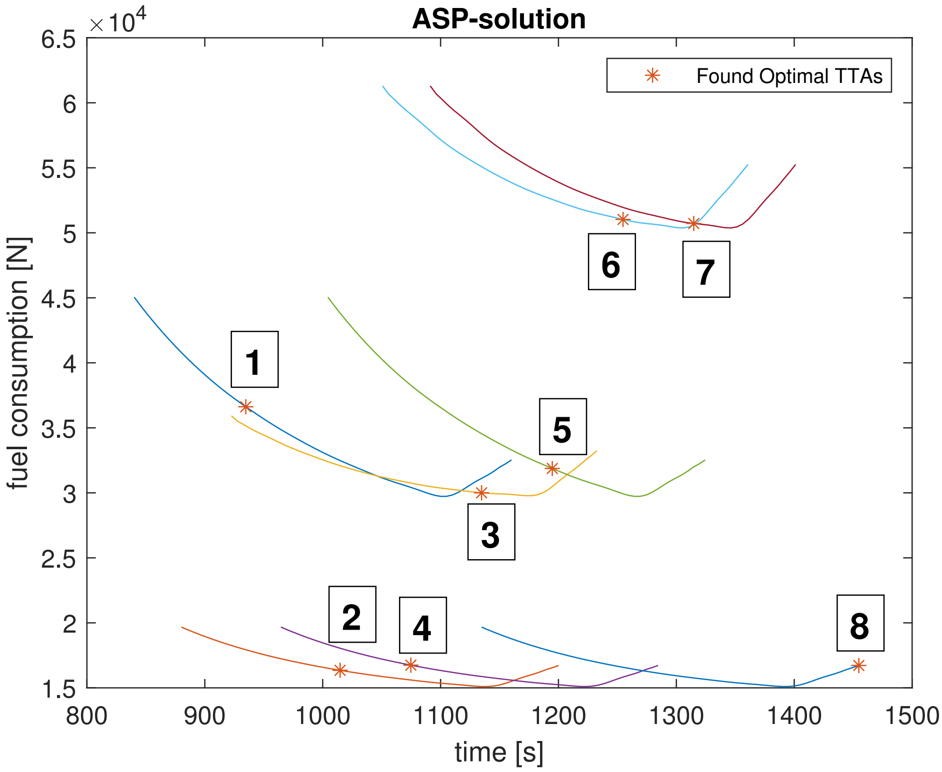


Figure 5.6: Minimum-fuel scheduling solution for a single (thus static) optimization run.

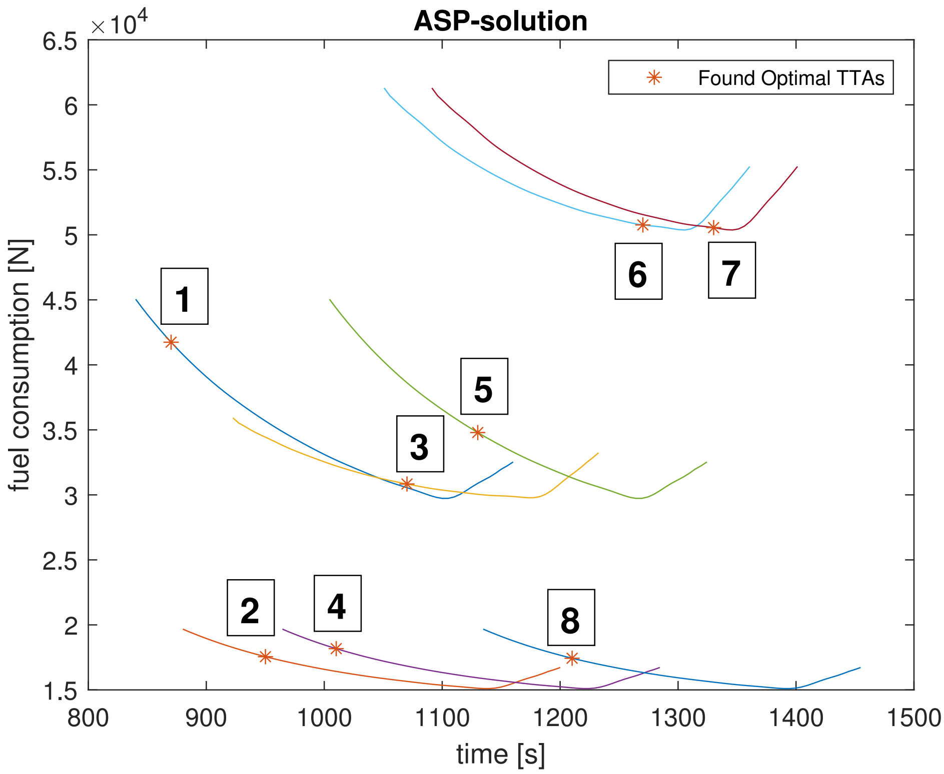


Figure 5.7: Minimum-delay scheduling solution for a single (thus static) optimization run. (Min-fuel post optimization enabled)

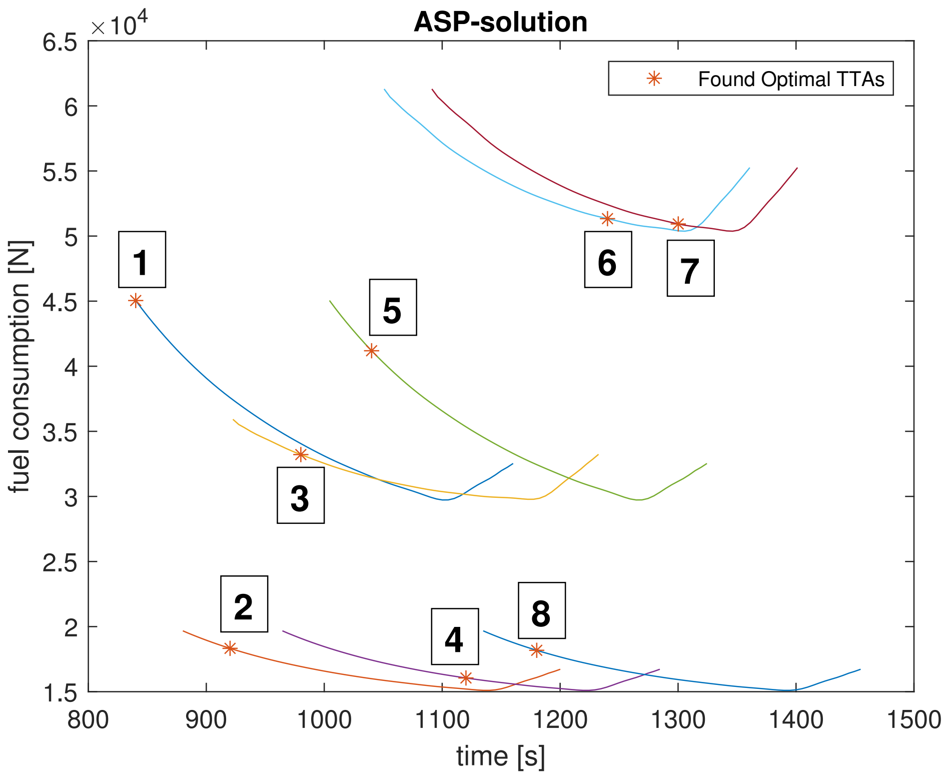


Figure 5.8: Minimum-time scheduling solution for a single (thus static) optimization run. (Min-fuel post optimization disabled)

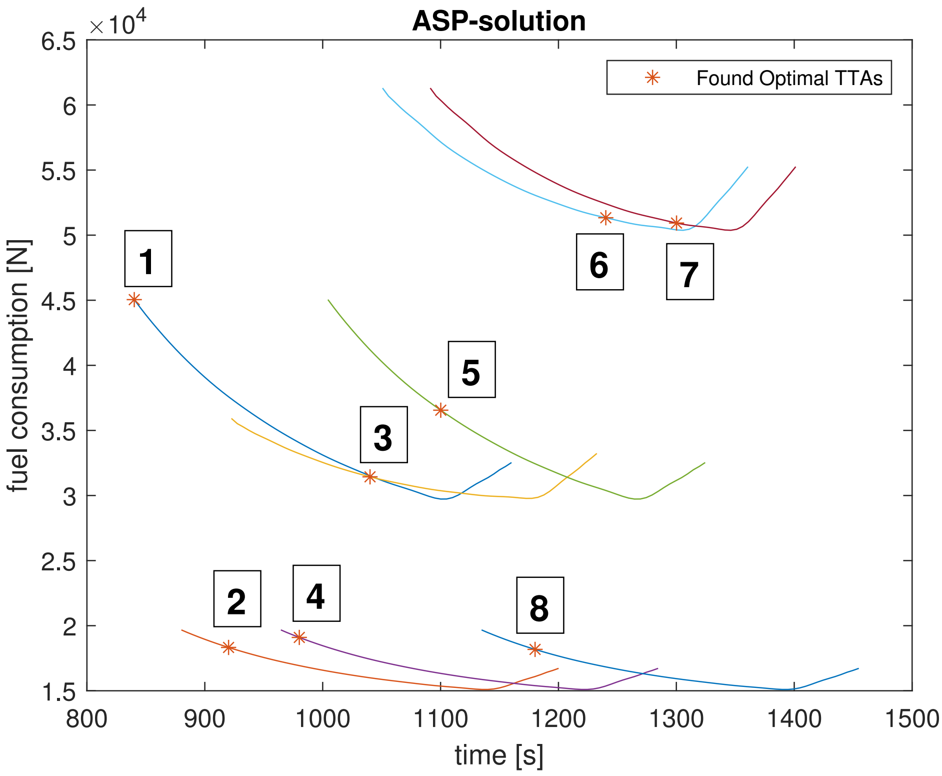


Figure 5.9: Minimum-time scheduling solution for a single (thus static) optimization run. (Min-fuel post optimization enabled)

5.2.2. THE STATIC ASP - NORMALISED FUEL CURVE BASED SCHEDULING

As discussed in section 5.1.3 there is an option to base the scheduling solutions on the normalised fuel curves. Although leading to less fuel-optimal scheduling solutions this would maintain a sense of fairness between different aircraft weight classes. This section briefly discusses how the fuel-minimal scheduling results change when computed based on normalised fuel curves. Figure 5.10 shows the minimum-fuel solution of the same static approach case that was tested in previous section in figures 5.6-5.9.

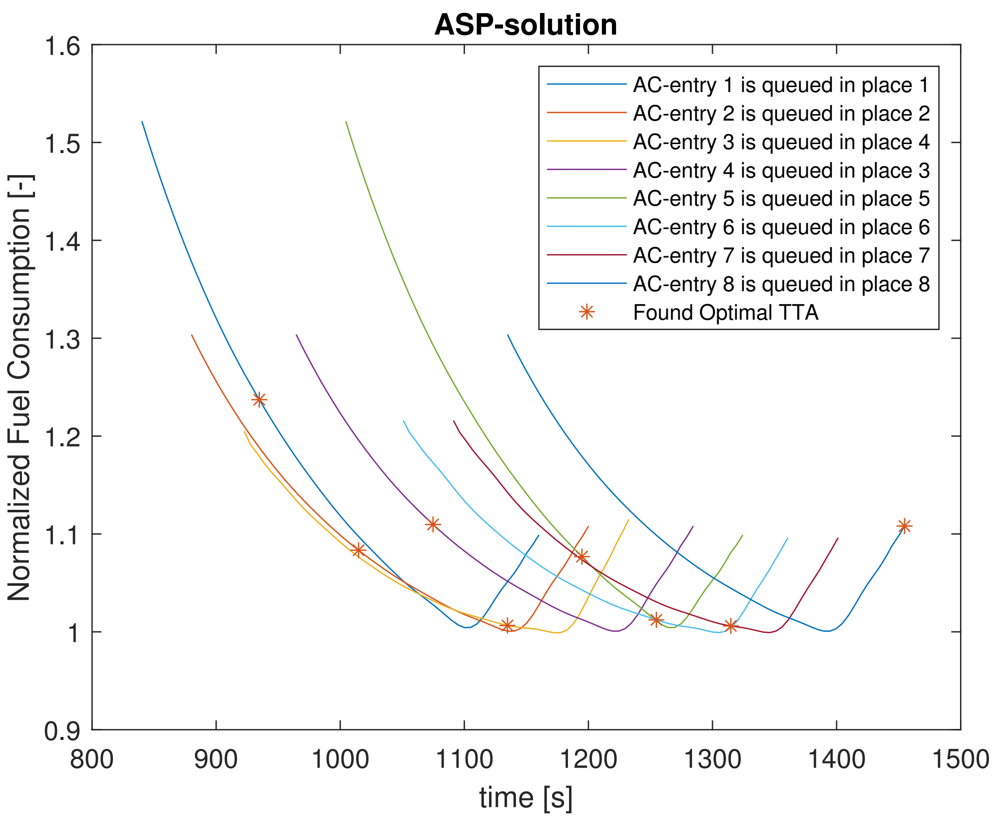


Figure 5.10: Minimum-fuel scheduling solution based on normalized fuel curves. (latest arrival: $1455s$; total fuel consumption: $25006kg$)

Comparing the scheduling solution based on the actual fuel curves in figure 5.6 to the solution based on the normalised fuel curves 5.10, no change is found, neither in the arrival sequence nor in the exact landing times within this sequence. However, one specific 8-aircraft approach case is insufficient to draw a decisive conclusion from. For this reason 100 unique static 8-aircraft approach cases were tested to compare the results using actual fuel curves and normalized fuel curves, that can be found in table 5.3.

Used Fuel Curves	**Average Fuel Consumption** [kg]	**Average Delay** [sec]	**Total Time** [min]
Actual Fuel Curves	2295.90 (100%)	8.3716 (100%)	842.505 (100%)
Normalized Fuel Curves	2310.67 (100.64%)	8.7588 (104.63%)	841.026 (99.82%)

Table 5.3: Minimum fuel scheduling performance based on actual- and normalized fuel curves for 100 static 8-aircraft approach cases and an average required runway throughput of 48 AC/hr

As expected, the total fuel consumption increases when using the normalized curves with respect to the actual fuel curve based scheduling. Surprisingly, this increase in fuel consumption is only 0.64%. The average delay increased by almost 5%. Although no conclusions can be drawn from this since only static cases were tested, it could be interesting to see how this affects the scheduling performance of dynamic test cases due to the propagation of delay. Time-wise no significant difference can be observed between using normalized- and actual fuel curves.

However, most interesting of applying normalized fuel curves when scheduling for minimum fuel is to see how this affects the fuel consumption amongst the individual weight classes by moving the scheduling priority away from the heavier aircraft and basically treating all aircraft types equally. This change in priority can be observed in figure 5.11 below, where for all 5 weight classes the increase in fuel consumption with respect to the aircraft's fuel-optimal landing time is plotted. Using the actual fuel curves for scheduling one can see that the fuel consumption of lighter aircraft increases significantly more than for heavy aircraft. Normalizing the fuel curves leads to a more equal increase in fuel consumption amongst all weight classes.

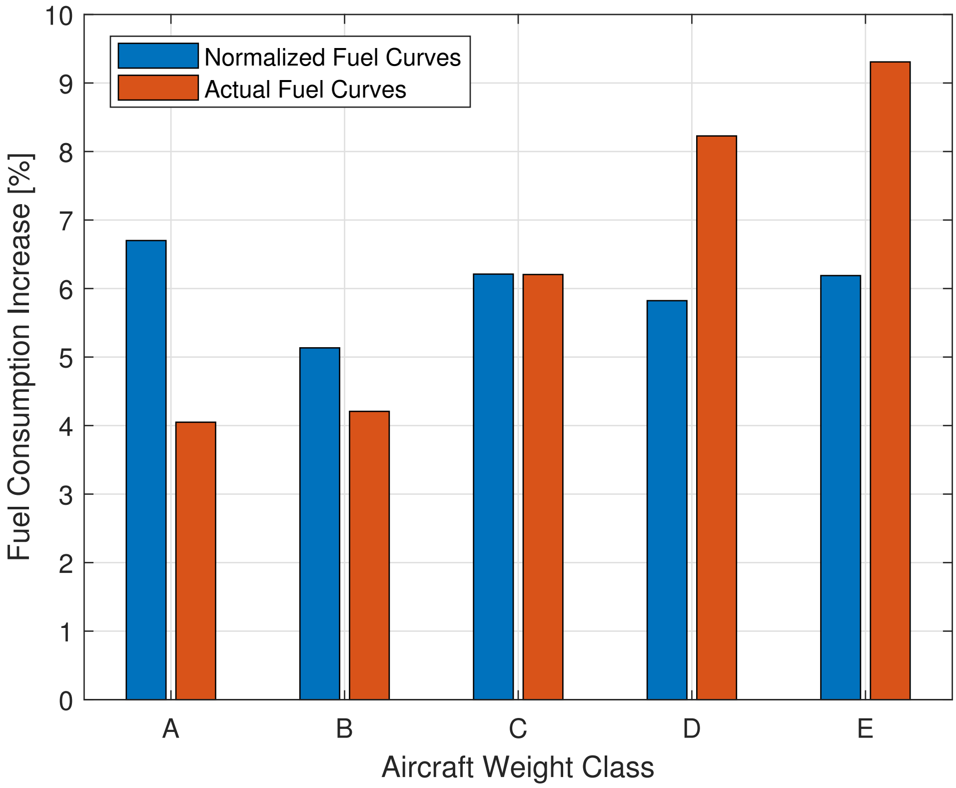


Figure 5.11: Fuel consumption increase with respect to minimum fuel landing times - Normalized Fuel Curves and Actual Fuel Curves

5.2.3. THE DYNAMIC ASP - SCHEDULING HORIZON SIZE

Now that the proper working of the scheduling algorithm in Module-2 has been verified by analysing the results of static approach cases, dynamic test cases can be tested in order to make a decision on both, the scheduling horizon size and the maximum position shift (MPS) that will be used to compute optimal arrival schedules in Module-2. As explained in section 4.8.1, the scheduling horizon size is limited to a maximum of 8 aircraft since a larger scheduling horizon would exceed the available system memory. The minimum tested scheduling horizon size of 4 aircraft is because this is the minimum required to allow for a MPS of 3 places.

As explained in section 4.8.1 the optimization performance of different scheduling horizon sizes can best be compared by finding the maximum achievable average required runway throughput. As mentioned earlier, the arrival schedule must be achievable without the need for aircraft holding. Since computation times are found to be very acceptable, even for a scheduling horizon size of 8 or 9 aircraft, the scheduling performance with respect to the achieved throughput is decisive in choosing a scheduling horizon size for the ASP-solver (i.e. Module-2). This was tested by running Module-2 in a dynamic environment with approach cases of 200 aircraft, while incrementally increasing the average required throughput by 0.6AC/hr whenever 5 consecutive test runs were completed successfully.

The results in figure 5.12 show that the highest average required throughput of 48AC/hr was achieved using an 8-aircraft scheduling horizon with a MPS of three positions. For this reason it was decided to use an 8-aircraft scheduling horizon with a 3-position MPS for the schedule optimization in Module-2.

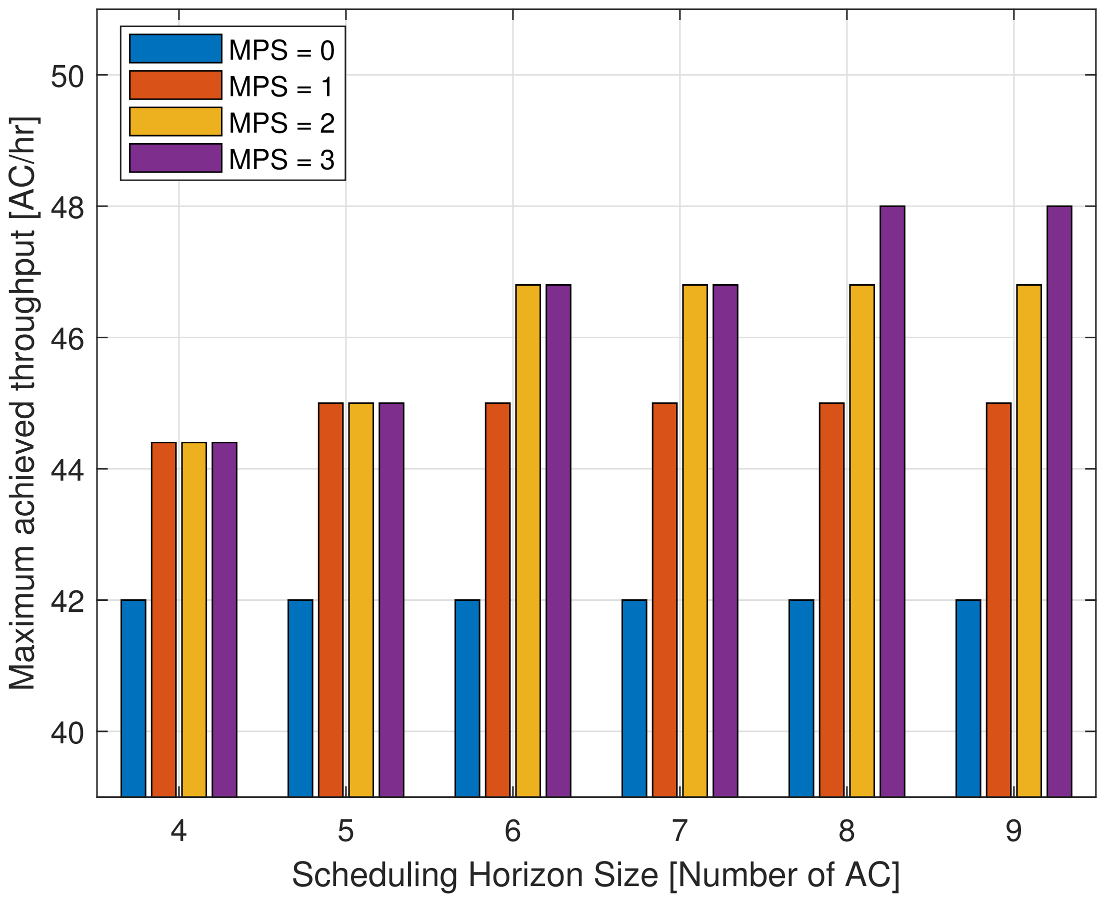


Figure 5.12: Maximum achieved average throughput level by the ASP-solver for various aircraft pool sizes and maximum position shifts (MPS) for 5 different weight classes using the same runway

The results in figure 5.12 also clearly show the potential increase in effective runway capacity that can be gained by optimal arrival scheduling with respect to the currently used FCFS scheduling strategy. It was found that the maximum achievable average throughput of the traditional FCFS scheduling strategy (i.e. MPS=0: no position shifting is allowed) is around 42AC/hr. Naturally, the scheduling horizon size does not affect this value since no position shifting is allowed. This means that ultimately, by using this arrival scheduling tool a 14.3% higher runway throughput can be achieved than when using the FCFS strategy.

However, it has to be noted that this is only an estimation since the maximum achieved throughput is very case specific when aircraft holding is not allowed. Actual throughput levels vary within a case since aircraft do not enter the TMA in fixed time intervals and much higher throughput levels can be achieved temporarily. Just as an example, the static approach case that was tested in section 5.2.1 managed to land 8 aircraft within the time interval of 41 – 336 seconds, which comes down to a required throughput of 1.63 aircraft per second. This also clearly points out the difference between static and dynamic approach cases and shows the importance of considering the effect of delay propagation.

This section will examine the scheduling performance of the different scheduling objectives in a dynamic optimization environment. The performance of the scheduling objectives can best be compared at the highest throughput level that is commonly achievable by all objectives since optimal scheduling is more effective at higher traffic density levels. For this reason it is important to first find out what the theoretical limit of the achieved runway throughput is for each of the scheduling objectives. Finally, it will be examined whether it is possible to increase the scheduling performance by applying dynamic scheduling objectives aiming to choose the right optimization objective before each iteration based on current throughput requirements.

5.2.4. THE DYNAMIC ASP - THEORETICAL THROUGHPUT LIMITS

The theoretical required throughput limit for each of the scheduling objectives in a dynamic optimization environment is plotted in figure 5.13. As was established in previous section, section 5.2.3, a scheduling horizon size of 8 aircraft was applied with a maximum allowed position shift of 3 places in the landing queue. Again, a throughput level is considered achievable if the optimization tool is able to schedule at least 200 aircraft without the need for aircraft holding for each of 5 consecutive test runs, after which the average required throughput is increased by $0.6AC/hr$. The achieved throughput of the 3 scheduling objectives is also compared to the First-Come First- Served (FCFS) scheduling strategy which was modelled as a minimum-fuel objective with a maximum position shift of $MPS = 0$.

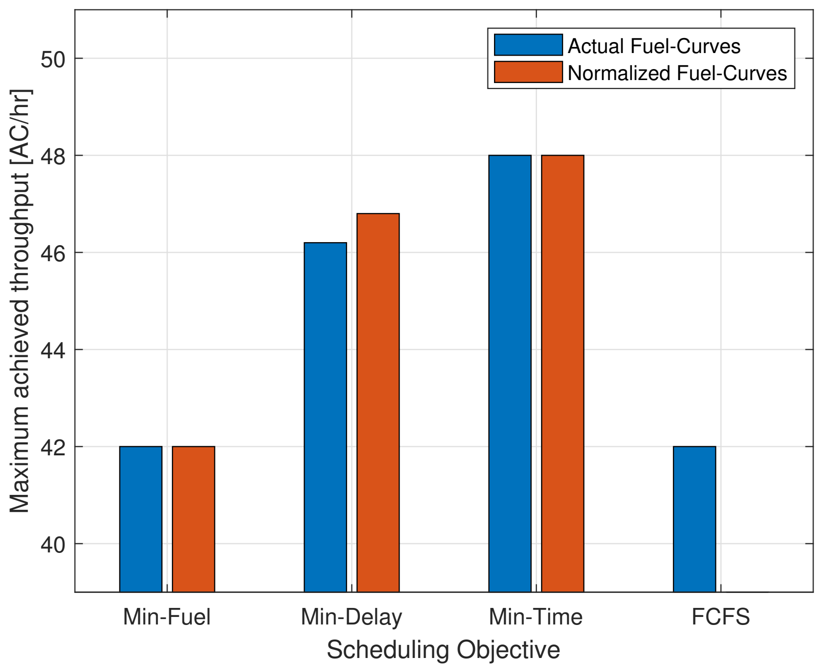


Figure 5.13: Theoretical Limits: Maximum achieved average throughput of each scheduling objective using actual- and normalized fuel curves. (SH-size: 8 aircraft, MPS: 3).

The results in figure 5.13 show that optimizing for minimum fuel achieved the lowest required throughput level of 42 AC/hr which is actually the same as the throughput level achieved by the FCFS scheduling strategy. This lies within the expected, since the minimum-fuel objective does not consider throughput at all and is solely focused on finding the most fuel-optimal landing schedule. For the minimum fuel objective, using normalized fuel curves has no effect on the achieved throughput level. As expected and tested before in figure 5.12, the minimum time objective achieved the highest required throughput of 48 AC/hr. Again, using normalized fuel curves does not affect the achieved throughput. The minimum delay objective performs more average with an achieved throughput level of 46.2 AC/hr. Interestingly, using normalized fuel curves increases the achieved throughput to 46.8 AC/hr. However, this performance increase is likely a coincidence and caused by an unfortunate combination of aircraft entries that could not be solved due to earlier made scheduling choices. Apart from this it would have been much more likely to see different results between the actual- and normalized fuel curves of the minimum fuel and minimum time scheduling objectives, since these objectives do actually prioritize between aircraft weight classes whereas the minimum delay objective does not. For these reasons it can be concluded that using normalized fuel curves will not significantly affect the maximum achievable throughput level of the scheduling objectives.

5.2.5. THE DYNAMIC ASP - SCHEDULING PERFORMANCE

Now that the theoretical throughput limits are known for each scheduling objective, the scheduling performance of these objectives can be compared at the highest commonly achievable required throughput, which is 42 AC/hr. The results of this comparison are given in table 5.4. Same as for the static case, the total fuel consumption and the average delay is compared. However, since this test case is a dynamic case it makes no sense to compare the last arrival times because new aircraft are continuously arriving. For a dynamic case it is only of importance whether the required throughput was achieved without accumulating and propagating delay. Further, the number of position changes in the landing sequence is compared between the scheduling objectives. This number represents the amount of times an aircraft was scheduled to land in a different order as it entered the freeze horizon. Since positions changes require two or more aircraft to pass each other during their final descent, the number of position changes is the most important indicator for the amount of conflicts that are to be expected. Since resolving trajectory conflicts increases the fuel consumption, this must be taken into account when choosing a scheduling objective once conflicts will be accounted for as well during the combined optimization of the ASP and the ATOP in section 5.4.

Scheduling Objective	*(Applied fuel curves)*	**Total Fuel Consumption** [kg]	**Average Delay** [s]	**Number of Sequence Position Changes** [-]
FCFS	*n.a.*	3.007e+07 (100%)	0.040	0
Minimum Fuel	*Actual fuel curves*	2.670e+07 (88.8%)	6.369	87
	Normalized fuel curves	3.010e+07 (100.1%)	1.288	184
Minimum Delay	*Actual fuel curves*	2.742e+07 (91.2%)	0.249	243
	Normalized fuel curves	3.109e+07 (103.4%)	0.000	162
Minimum Time	*Actual fuel curves*	2.953e+07 (98.2%)	0.864	311
	Normalized fuel curves	3.175e+07 (105.6%)	0.257	131

Table 5.4: Scheduling performance using various optimization objectives (five 200-aircraft dynamic approach cases were tested with an average required throughput of 42 AC/hr)

The results in terms of fuel consumption are within the expected. Interesting is that the FCFS schedule consumes even more fuel than the minimum time scheduling objective. However, since the FCFS schedule does not perform any sequencing changes, the expected lower number of conflicts will compensate for this to some extend.

The average delay is negligible for the minimum time and minimum delay objectives. Only when scheduling for minimum fuel using the actual fuel curves the average delay is a little higher which is due to the fact the heavier aircraft are forced to land exactly at their fuel-optimal landing time and therefore lighter aircraft have to be moved to allow for this and are likely to be delayed in the process.

Lastly, there is an interesting difference in the behaviour the required number of position changes between the 3 scheduling objectives. While applying normalized fuel curves during fuel-minimal scheduling doubles the number of position changes, applying normalized fuel curves for post-optimization of the delay and time objectives tends to decrease the required position changes. The reason for this is that applying normalized fuel curves only removes the fuel-based priority that is given to the heavier weight class aircraft. When scheduling for minimum fuel, this results in more scheduling options since the optimal schedule is no longer dictated by the largest aircraft. However, the minimum time and minimum delay scheduling objectives prioritize between weight classes based on their required time-based wake separation. Therefore, applying normalized fuel curves does only affect the scheduling solution when post-optimizing for minimum fuel. The reason why the required position changes are actually lower when applying normalized fuel curves for the minimum time and minimum delay objectives is because these time-based objectives tend to prioritize lighter aircraft before heavier aircraft to reduce wake separation times. Since this priority is exactly the opposite of the fuel-based scheduling priorities, removing the fuel-based priorities by applying normalized fuel curves prevents excessive position changes trying to meet both priorities.

5.2.6. THE DYNAMIC ASP - DYNAMIC SCHEDULING OBJECTIVES SWITCHING

As explained in section 4.8.2, it makes sense to not use the same optimization objective for each optimization iteration during the entire aircraft approach case. By choosing the minimum fuel scheduling objective whenever the available capacity allows for it and switching to the minimum time objective whenever this is necessary to cope with the required throughput level, one could schedule for minimum fuel but still provide an option to overcome temporary traffic peaks.

As mentioned before in section 4.8.2, this section will test two different strategies that define the conditions that must met deciding whether to apply the fuel-minimal or the time-minimal scheduling objective. Both dynamic scheduling strategies will be tested for average required throughput levels above the theoretical throughput limit of the minimum fuel scheduling objective and up to the theoretical limit of the minimal time scheduling objective. Therefore the tested average throughput will range from $42.6AC/hr$ to $48AC/hr$, as was determined in section 5.2.4 using the same dynamic 200-aircraft dynamic approach cases for the tested throughput levels as those that were used to determine the theoretical throughput limits.

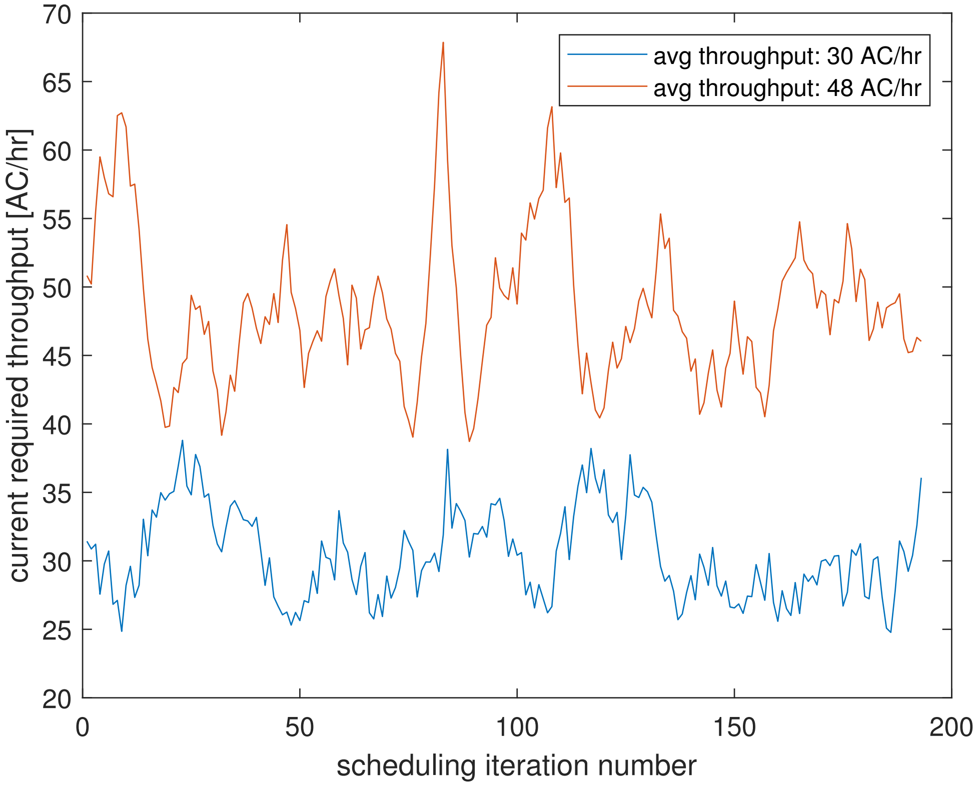


Figure 5.14: Example of currently required throughput for each scheduling iteration over a 200-aircraft scheduling test case (Based on uniformly distributed entrance times, as described in section 4.3)

- **Strategy 1 - Based on currently incoming throughput**. The first strategy chooses between fuel-optimal scheduling and time-minimal scheduling based on the throughput that is required by all aircraft that currently are within the scheduling horizon. An example of this required throughput over an entire 200-aircraft scheduling test case is shown in figure 5.14. Since the theoretical throughput limit of the minimum fuel objective (limit = $42AC/hr$) has been determined in section 5.2.4, one might be able to predict whether this objective will be able to cope with the required throughput. In order to do so a throughput threshold level (TTL) must be determined at which the scheduling objective is switched to time-minimal scheduling as soon as the required throughput exceeds this TTL. Since it is very likely that, in order to prevent the build-up of delay, time-minimal scheduling must be applied even before the fuel-minimal limit threshold of $42 AC/hr$ is reached, TTLs ranging from $36-42 AC/hr$ will be tested.

- **Strategy 2 - Based on expected delay**. The second strategy is delay based. Theoretically, as long as delay does not propagate on to the next incoming aircraft (an aircraft that has not yet entered the scheduling horizon), which would lead to reduced optimization potential for the next scheduling iteration, it is perfectly fine to schedule for minimum fuel, even during temporary traffic peaks. Therefore this strategy will apply time-minimal scheduling whenever delay propagation is detected. This is done by scheduling for minimum fuel first and then re-scheduling for minimum time only when the last aircraft entering the freeze horizon is scheduled to be delayed.

The scheduling performance in terms of fuel consumption of these two dynamic scheduling strategies are plotted in figure 5.15 and are compared to the scheduling results when solely running the time-minimal scheduling objective over the entire approach case. Note that the legend of this figure also displays whether the applied scheduling strategy managed to schedule all aircraft without the need for aircraft holding.

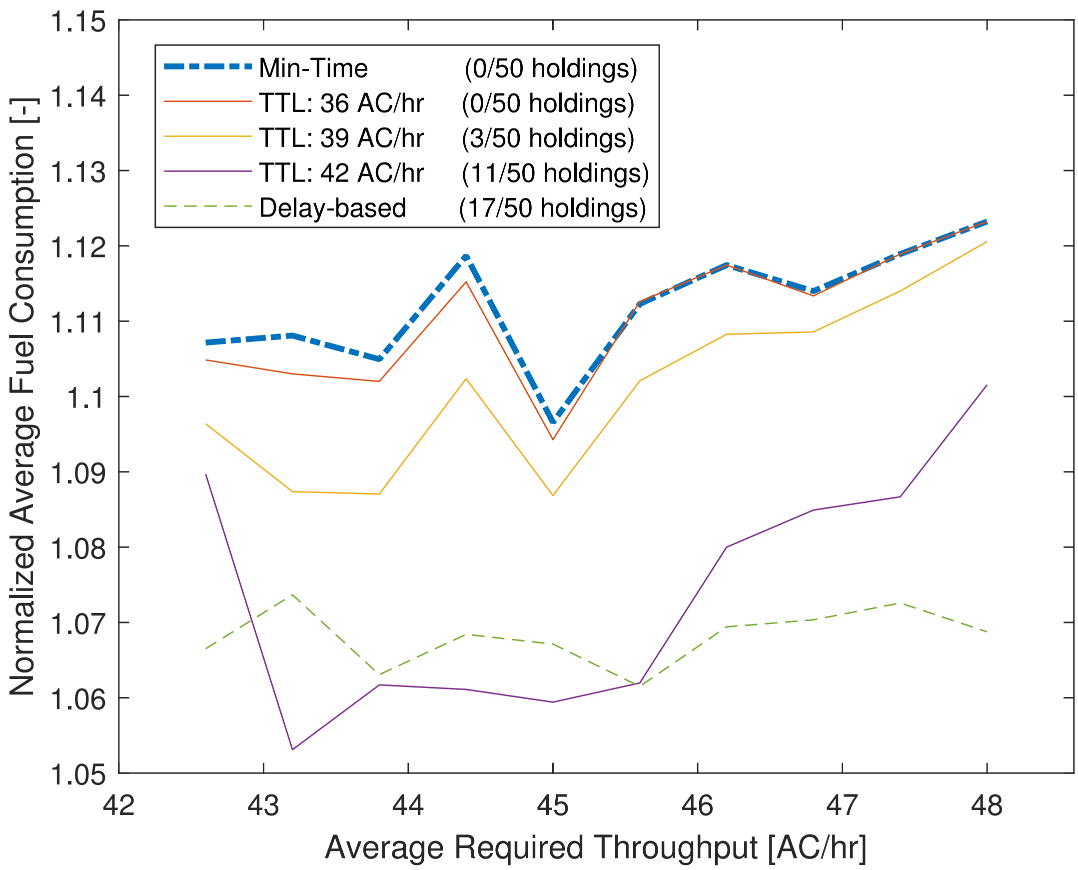


Figure 5.15: Fuel-performance comparison of 2 dynamic scheduling strategies for increasing average required throughput. (5 200-aircraft dynamic approach cases tested per throughput level)

Before comparing the dynamic objective strategies to each other it must be noted that although the average fuel consumption increases with the average required throughput, it does not do so consistently. The minimum time solution, for example, shows a significantly higher average fuel consumption at $44.4 AC/hr$

than at 45.0AC/hr. The reason for this is that this fuel-performance is very case specific. Referring back to figure 5.14, it can be noticed that the current required throughput differs a lot over the time of the test case. Since most fuel is "lost" during the high traffic peaks within a test case, test cases with more extreme peaks tend to cost more fuel than test cases with more constant required throughput, even if the average required throughput is lower. With that said, it can be expected that the average fuel consumption will increase more consistently with the required throughput if many more test cases would have been evaluated per throughput level.

Starting off with the delay-based strategy 2, it is clear that this dynamic objective scheduling strategy did not work as intended. As can be seen in the legend of figure 5.15, 17 out of the 50 200-aircraft approach cases had to be aborted because the scheduling strategy could not find a feasible landing time solution for an aircraft, which would require the aircraft to be put in holding. The reason why this strategy was unable to solve some of the approach problems is because delay propagated already earlier within the scheduling horizon and fixed in the previous scheduling iterations. Often, the minimum time objective will not be able to remove the delay within one scheduling iteration. Unfortunately, early in the scheduling horizon it is very hard to distinguish between delay caused by delay propagation and delay caused by objective based scheduling decisions. Therefore it can be concluded that a delay-based dynamic scheduling strategy is consistent and a more predictive strategy is required.

Strategy 1, the dynamic scheduling strategy that bases its scheduling objective decisions on the currently required throughput is a much more predictive strategy, especially for lower throughput threshold levels (TTLs). As can be seen in figure 5.15, choosing $TTL = 36AC/hr$ results in a dynamic scheduling strategy that manages to consistently schedule all of the 50 approach cases. Although much more consistent than the delay-based strategy, increasing the TTL above 36AC/hr again leads to infeasible scheduling solutions. Interesting to see is how the fuel benefit of dynamic scheduling objectives decreases for increasing required throughput. The higher the average throughput, the less often the currently required throughput in the scheduling horizon will drop below the defined TTL, which means that at some point only the minimum time objective will be applied. For $TTL = 36AC/hr$ one can clearly see how the fuel consumption of the dynamic objective equals the fuel consumption of the fixed minimum time objective from average throughputs of 45.6AC/hr and upwards.

To conclude, by switching from fuel-optimal scheduling to the minimum time scheduling objective whenever the incoming throughput exceeds a threshold of 36AC/hr a slight decrease in fuel consumption was achieved with respect to scheduling for minimum time only. More significant fuel benefits seem to be possible, however a more sophisticated strategy is required to avoid infeasible scheduling solutions more consistently.

5.3. CONFLICT RESOLUTION

Now that Module-1 and Module-2 have been tested and optimal tool settings, in terms of scheduling horizon size, maximum position shift and throughput threshold level, have been established, Module-3 will be tested. This section will investigate how conflicts are resolved, how often conflicts actually occur and how much the resolution of conflicting trajectories affects the fuel consumption. The results obtained in the section are based on dynamic objective scheduling with the TTL set to $36AC/hr$ as explained in previous section.

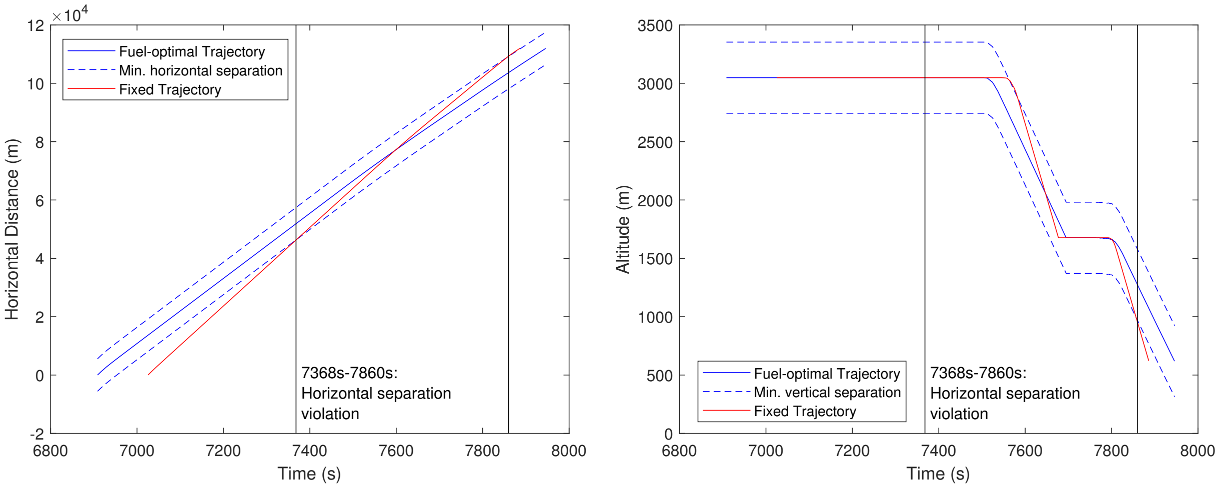
5.3.1. CONFLICT RESOLUTION

This section presents a typical approach trajectory conflict between two aircraft and shows how module-3 resolves this conflict. Since conflicts are caused by position changes within the landing sequence and approach trajectories are computed and fixed in order of this landing sequence, almost all conflicts that occur are between an already fixed approach trajectory and an aircraft that is scheduled to land later but will enter the freeze horizon at an earlier point in time. This means that the the approach trajectory that is still open must find a way to let the aircraft with the fixed trajectory pass in order for it to land at an earlier time. Therefore the open trajectory aircraft will have to descend to a lower altitude earlier than it would do following its fuel-optimal trajectory. This way the trajectory will provide sufficient vertical separation for the fixed trajectory aircraft to pass during the time interval in which the horizontal separation is violated due to overtaking.

Figure 5.16 shows this typical trajectory conflict by plotting the track distance and altitude at each point in time respectively. As can be noticed in figure 5.16(a), the horizontal separation is violated between 7368-7860 seconds due to the overtake. At the same time, figure 5.16(b) shows that during this entire interval vertical separation is violated as well and thus a trajectory conflict exists. Since the overtake manoeuvre and the consequential horizontal separation violation is mandatory in order to meet the computed landing schedule, the vertical separation violation must somehow be resolved.

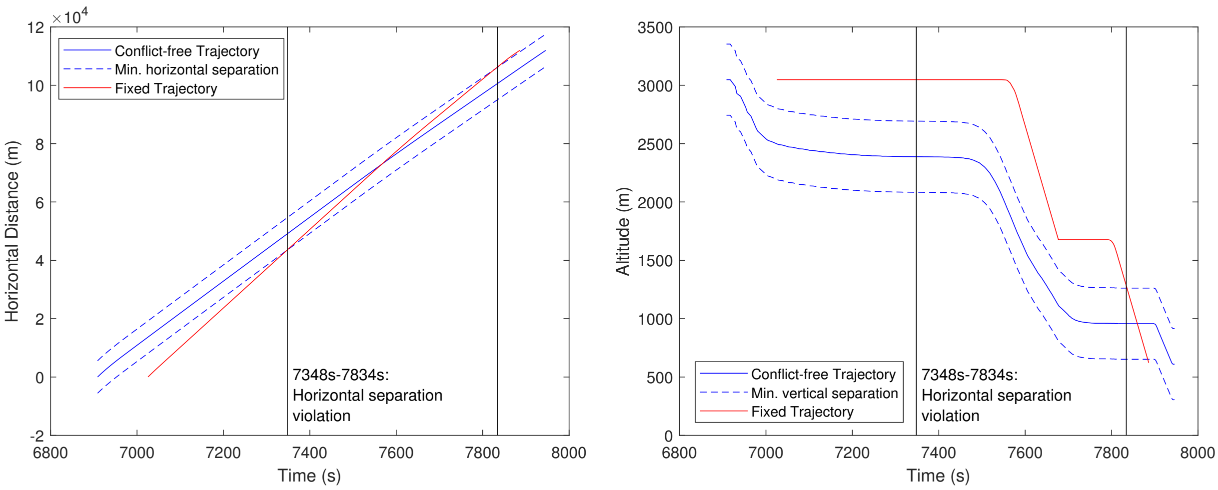
Figure 5.17 shows how this vertical separation is maintained to resolve the trajectory conflict. As expected, the descend occurs directly after entry into the freeze horizon. Interesting is that the computed fuel-optimal conflict-free trajectory loses more altitude than necessary to maintain vertical separation. The reason for this is that due to the loss of altitude the interval in which the horizontal separation is violated also moves forward in time by a few seconds. This was necessary to allow for the vertical separation violation that has to take place in the last part of the descent.

To allow for an easier visualization of the found trajectories, a 2-dimensional plot over time of the conflict-neglected trajectory and the conflict-free trajectory is shown in figures 5.18 and 5.19 respectively.



(a) track distance (b) altitude profile

Figure 5.16: Typical conflict between fixed trajectory (red) and planned fuel-optimal approach trajectory (blue)



(a) track distance (b) altitude profile

Figure 5.17: Conflict Resolution by early loss of altitude

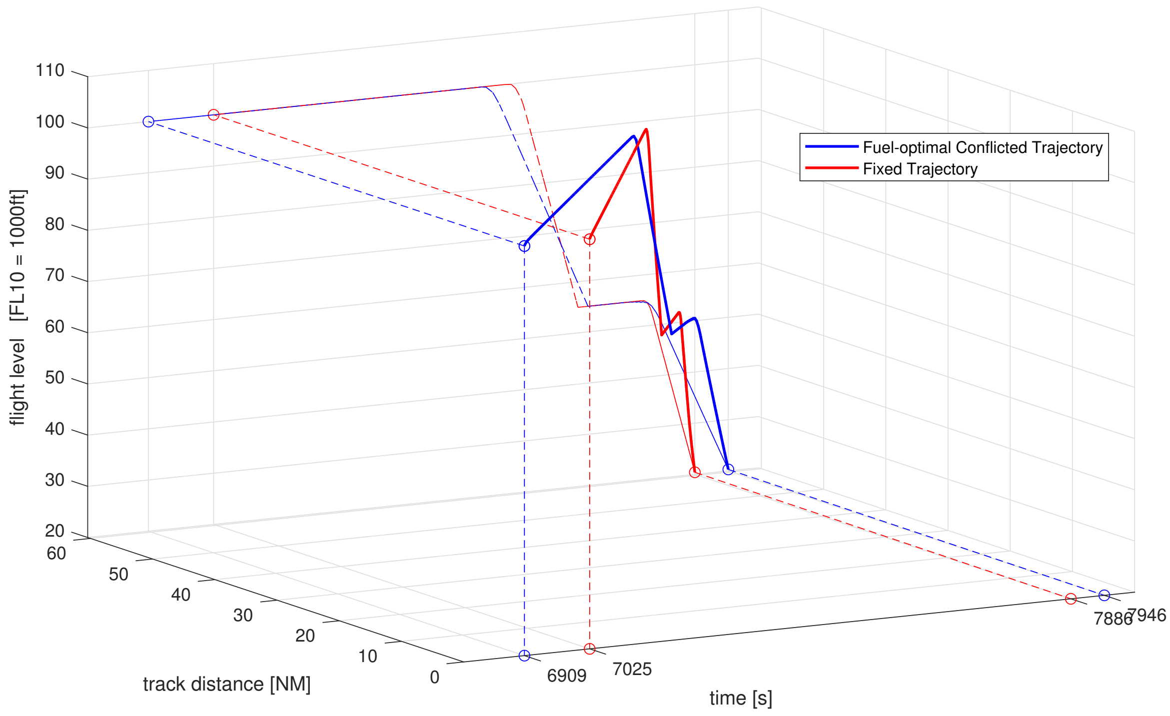


Figure 5.18: 2-dimensional fuel-optimal conflicted trajectory over time

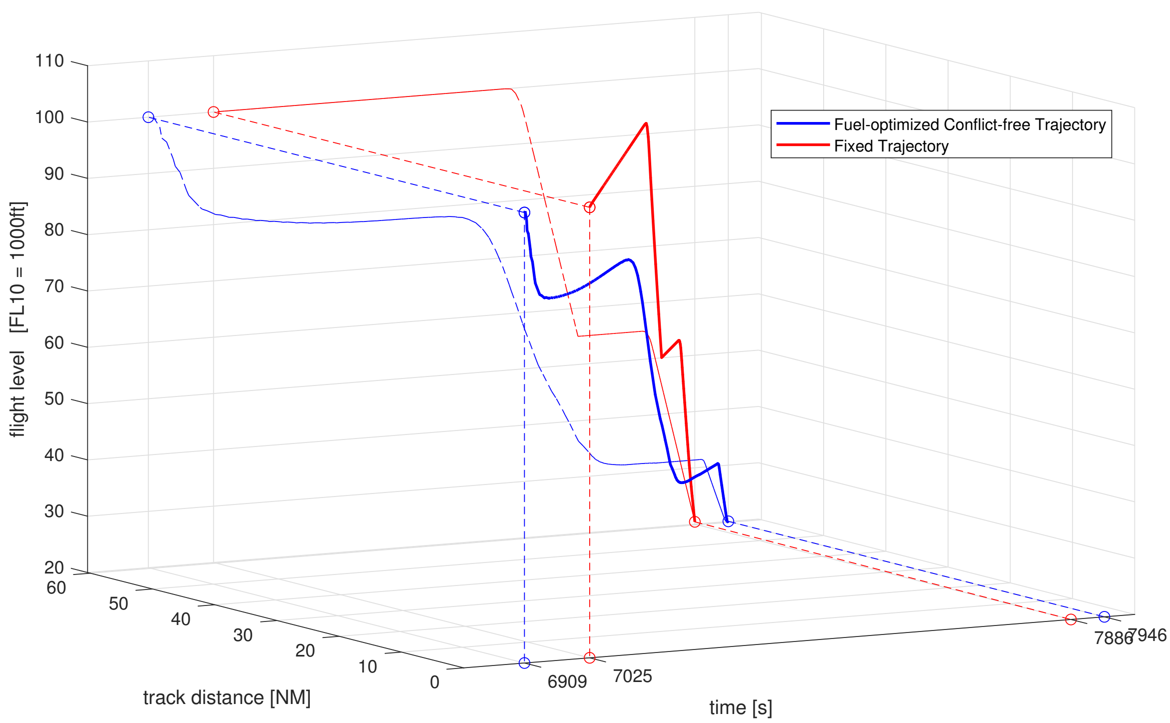


Figure 5.19: 2-dimensional fuel-optimized conflict-free trajectory over time

5.3.2. EFFECT OF CONFLICTS ON FUEL CONSUMPTION

Figure 5.20 shows how many of the scheduled aircraft have conflicting approach trajectories that have to be resolved for various levels of required average throughput. Together with the average fuel cost of resolving a conflict, this information gives an estimation of how much the caused conflicts will affect the fuel-performance of the scheduling solutions. This is tested to make sure that the fuel consumption benefit obtained by optimal arrival scheduling is not negated by the caused conflicts.

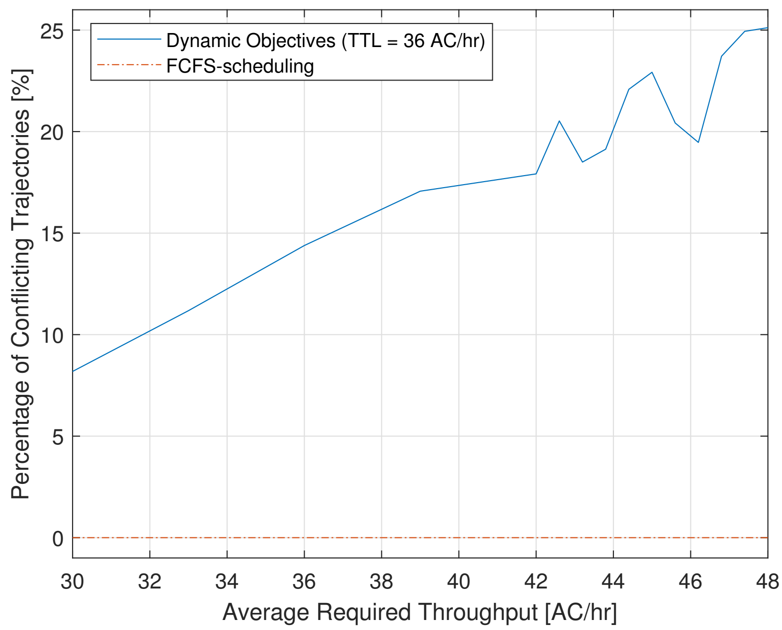


Figure 5.20: Percentage of conflicting approach trajectories, for various required average throughput levels (SH-size: 8ACs, MPS: 3, TTL: $36AC/hr$)

Figure 5.20 shows that the average amount of trajectories that need to be altered to resolve conflicts increases with the level of required throughput and ranges between 8% and 26%. Interestingly, FCFS-scheduling does never cause any conflicts, which means that conflicts are solely caused as a result to arrival sequencing. Figure 5.21 shows the average percentage of extra fuel consumption that is necessary to resolve a trajectory conflict. The average of 200 randomly encountered trajectory conflicts for each weight class was taken. The extra fuel consumption ranges from 3.2% to 5.4% depending on the aircraft weight class. As expected, the lower weight aircraft are more efficient at manoeuvring around a conflict than the higher weight classes.

Considering these results, when scheduling with dynamic scheduling objectives an average fuel consumption increase between 1-2% is expected after resolving conflicts. Therefore, referring back to earlier scheduling results in table 5.4, it can be expected that at a required throughput of $42AC/hr$ the fuel consumption of FCFS-scheduling will match the fuel consumption when scheduling using dynamic objectives. However, for lower levels of required throughput, dynamic objective scheduling will apply fuel-minimal scheduling more often which will not only result in more fuel-optimal scheduling results but will also cause less conflicts and thus lower the conflict-induced fuel consumption increase. A required throughput higher than $42AC/hr$ cannot be achieved by FCFS-scheduling as explained earlier.

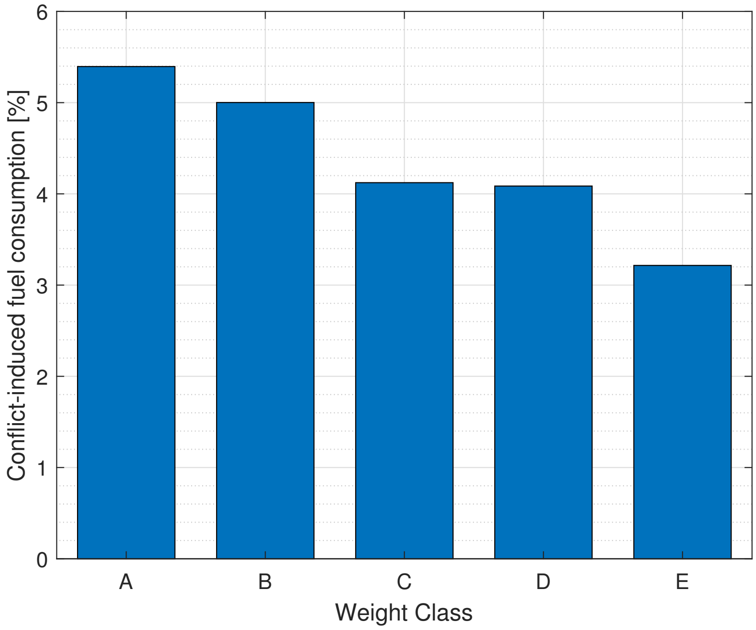


Figure 5.21: Average percentage of added fuel consumption due to conflict resolution for each weight class (SH-size: 8ACs, MPS: 3, TTL: $36 AC/hr$)

5.3.3. UNSOLVABLE CONFLICTS

Unfortunately, not all of the encountered trajectory conflicts could be resolved to fully meet the required minimum radar separation. In a few cases, the combination of close TMA entry times and similar weight classes of the corresponding incoming aircraft leads to landing times that are scheduled so closely together that the trajectory optimization module simply cannot find a conflict-free trajectory solution that still meets the scheduled landing times. Figure 5.22 shows the percentage of conflicts that could not be resolved for different levels of required throughput. As expected, the higher the required throughput, the more often the scenario explained above appears, leading to an unsolvable trajectory conflict. Unfortunately, this increasing behaviour of unresolvable conflicts in figure 5.22 is very inconsistent with respect to the corresponding required throughput. The reason for this is the limited amount of unresolvable conflict data from the performed approach test cases. Much longer approach cases must be computed in order to yield results that are statistically significant.

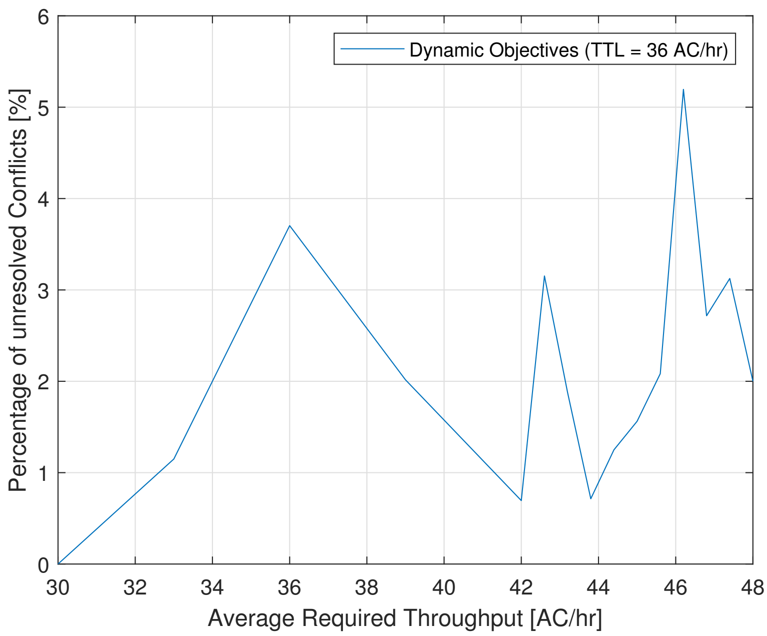


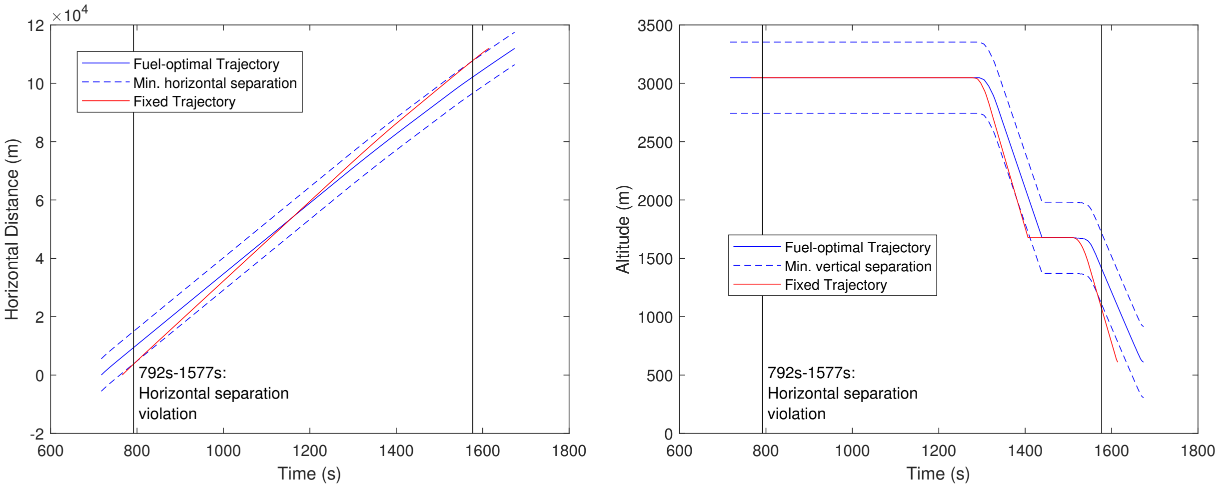
Figure 5.22: Percentage of Conflicts that could not be resolved, for various throughput levels

Figure 5.23 shows an example of a conflict that could not be solved. Again an approach trajectory must be found such that a following aircraft can overtake while respecting both, the separation distances and the scheduled times of arrival. As figure 5.23(a) shows, the fuel-optimal approach trajectory violates the horizontal separation to the fixed aircraft trajectory over almost the entire freeze horizon time span, since both the entry time and the arrival time are extremely close to each other.

The reason why this conflict cannot be solved lies mainly in the applied path constraints to ensure continuous descent approaches, explained in section 4.5.3. These constraints do not allow an aircraft to accelerate or climb. However, by acceleration the aircraft could generate enough of a time difference to lose altitude before the fixed aircraft trajectory enters the freeze horizon. Climbing would be another option to resolve the conflict by letting the other aircraft pass below instead of above. This latter option would prevent both aircraft to cross the same altitude at given point in time and would thus solve the horizontal separation violation problem. Figure 5.24 simulates this solution by allowing the aircraft to enter at a higher flight level to prevent crossing altitudes. As can be noticed, a continuous descent approach profile is found that maintains separation and meets the scheduled arrival time.

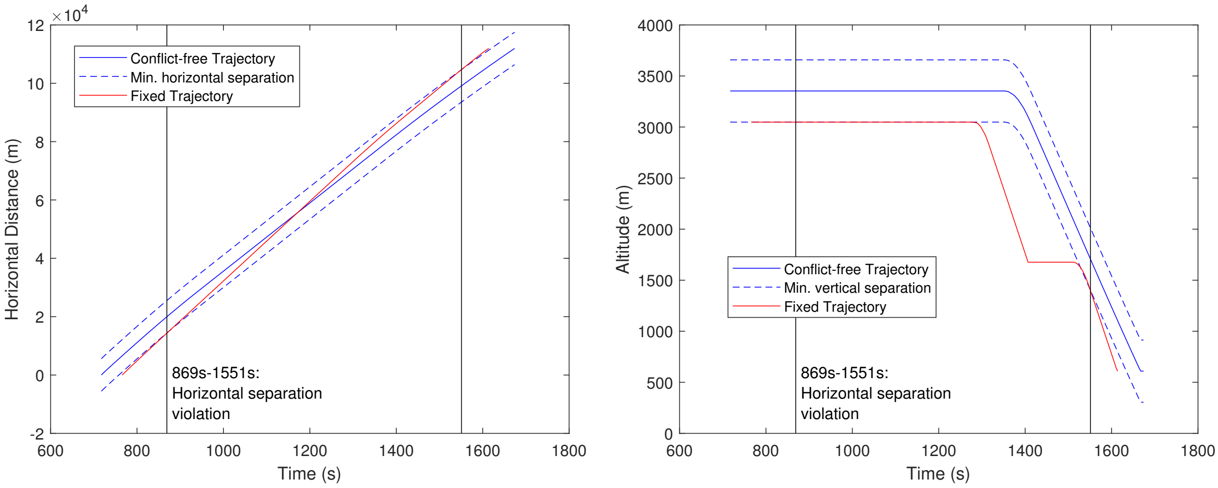
Although all unsolved conflicts could be resolved this way, it is far from optimal since this would mean that aircraft would occasionally have to climb within the scheduling horizon, which would impair continuous descent operations. For future research two far more fuel-optimal solutions come to mind. Firstly, conflict resolution should be done cooperatively. This would mean that before fixing the other aircraft's trajectory it could slow down just enough to provide space for the foregoing aircraft to lose altitude. Secondly, the freeze horizon should be enlarged which requires earlier arrival scheduling. This way aircraft would have more space and thus time to lose or gain distance and thus have more options the solve an occurred trajectory conflict.

Again, to allow for an easier visualization of the found trajectories, a 2-dimensional plot over time of the conflict-neglected trajectory and the conflict-free trajectory is shown in figures 5.25 and 5.26 respectively.



(a) track distance (b) altitude profile

Figure 5.23: Unsolvable conflict between fixed trajectory (red) and planned fuel-optimal approach trajectory (blue)



(a) track distance (b) altitude profile

Figure 5.24: Conflict Resolution by allowing freeze horizon entry at a higher flight level (+1000ft)

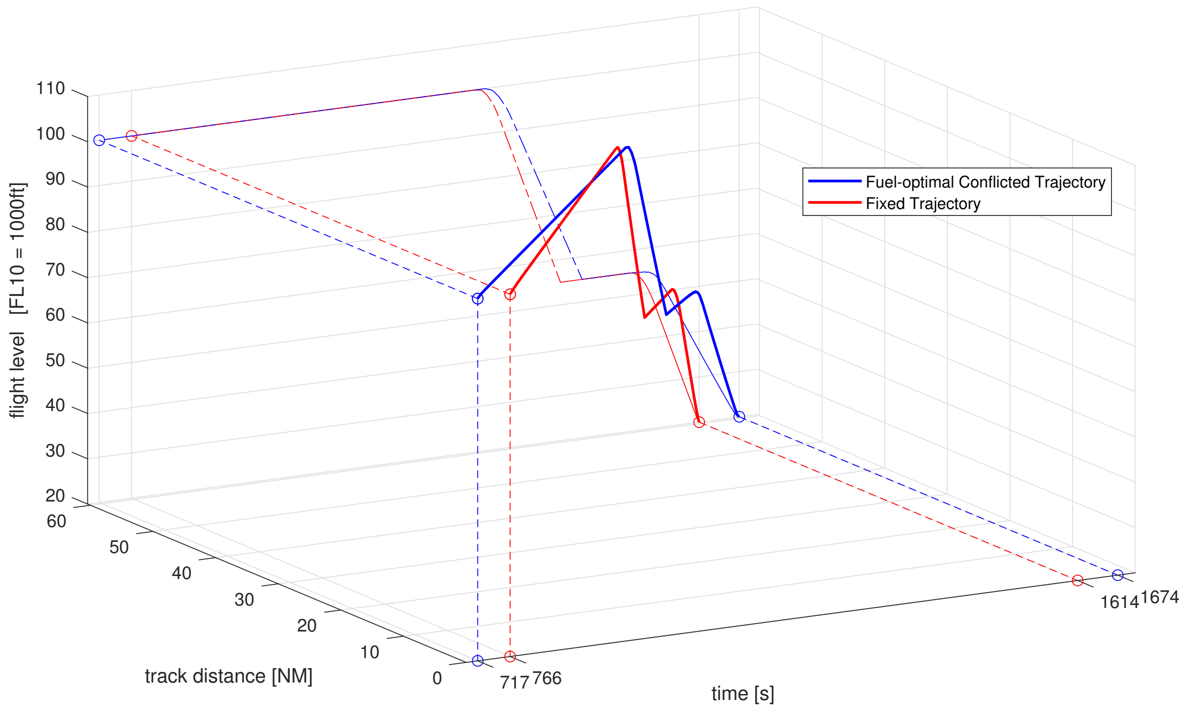


Figure 5.25: 2-dimensional fuel-optimal conflicted trajectory over time

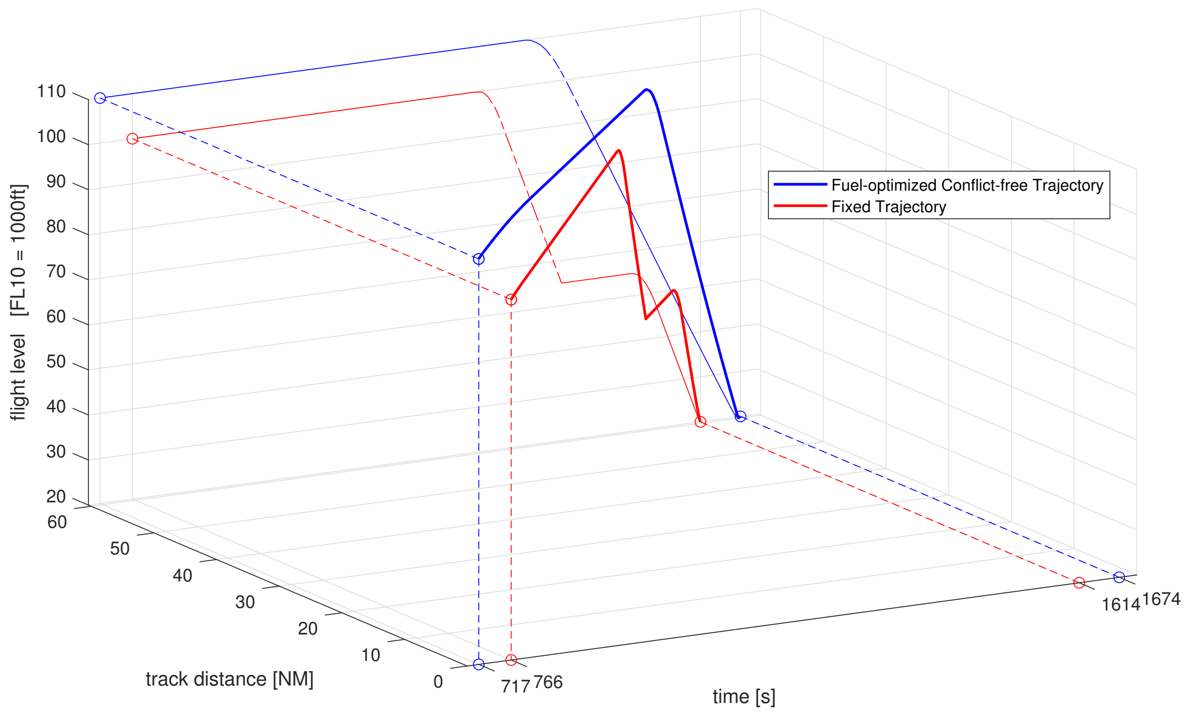


Figure 5.26: 2-dimensional fuel-optimized conflict-free trajectory over time

5.4. SIMULTANEOUS OPTIMIZATION OF ASP & ATOP

Finally, after verifying the correct working of the 3 individual models and finding optimal scheduling settings in terms of scheduling horizon size, maximum allowed position shift and throughput threshold level, it is time to run the developed optimization tool in its entirety. This section will present the results achieved by the simultaneous optimization of the arrival schedule and the corresponding approach trajectories, in terms of consumed fuel for a range of average required throughput levels. By comparing the results of the applied dynamic objective scheduling strategy to the results of fuel-minimal and time-minimal scheduling, the main research question can be answered as to whether a balance between time-optimality and fuel-optimality has been found for any given required runway throughput.

The following final optimization tool settings were established in previous sections:

- **Scheduling Horizon Size:**
  A fixed number of 8 aircraft are considered for scheduling for each optimization iteration.
- **Maximum Position Shift (MPS):**
  Arrival aircraft must land within a maximum of 3 positions of their respective place in the TMA entry sequence (i.e. the FCFS-sequence).
- **Throughput Threshold Level (TTL):**
  A TTL of 36AC/hr was found to achieve the most fuel-optimal scheduling results, while still allowing maximum throughput levels to be achieved.
  - **Current required throughput <= 36 AC/hr:** Fuel-minimal scheduling based on actual fuel curves is applied.
  - **Current required throughput > 36 AC/hr:** Time-minimal scheduling based on actual fuel curves is applied.

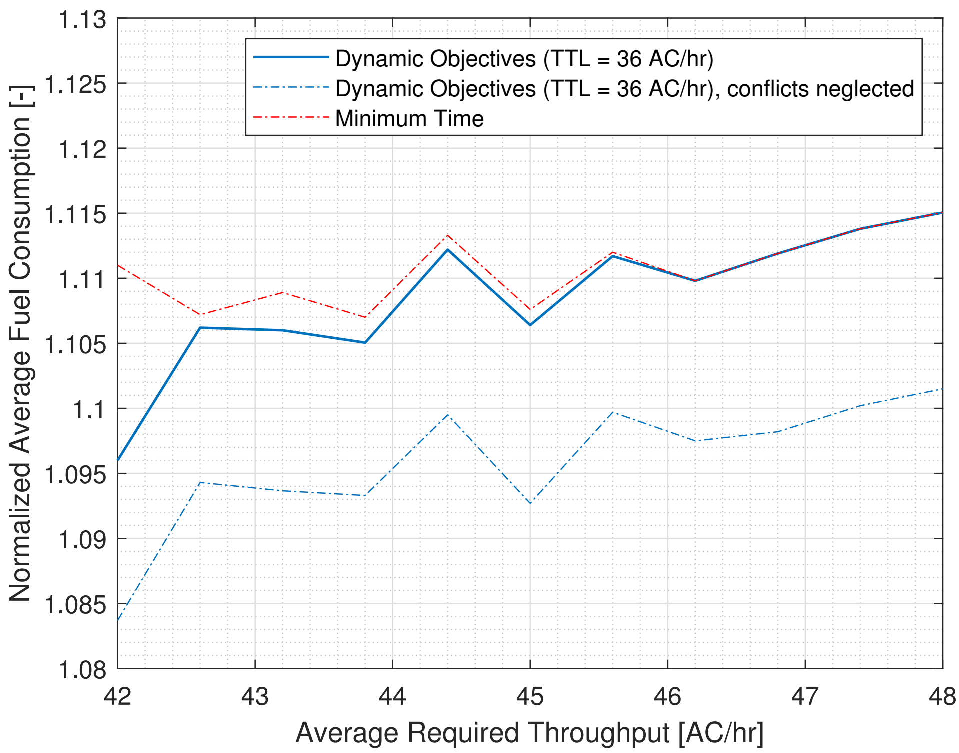


Figure 5.27: Effect of conflicts on average fuel consumption (SH-size: 8ACs, MPS: 3)

Before analysing and comparing the achieved results, it makes sense to shortly refer back to section 5.3.2 and find out how much the average fuel consumption is actually affected by resolving the caused trajectory conflicts. Figure 5.27 shows the difference in average fuel consumption between neglecting conflicts and resolving caused conflicts. As expected, resolving the conflicts between the scheduled approach trajectories increases the average fuel consumption around 1.2%. Although one would expect the fuel cost of resolving conflicts to rise with increasing throughput levels, no significant fuel cost increase can be noticed.

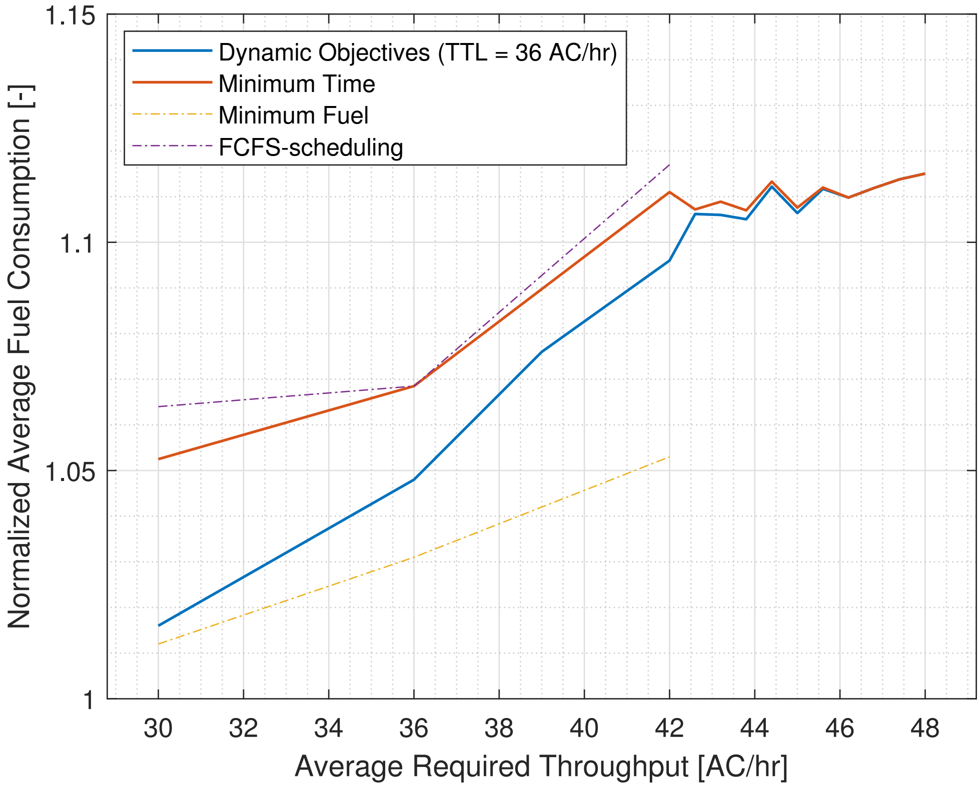


Figure 5.28: Fuel-optimality of the developed optimization tool for various levels of required throughput (SH-size: 8ACs, MPS: 3)

Figure 5.28 shows the average fuel consumption results that were achieved by the developed optimization tool, for a range of average required throughput levels. The plot also shows the solutions that would result from scheduling solely for minimum fuel or minimum time and the solution that would be achieved if no position changes within the landing sequence were allowed (i.e. the FCFS-sequence).

As was aimed for, the fuel consumption resulting from dynamic objective scheduling is close to the minimum fuel solution at lower throughput levels and slowly increases towards the minimum time solution with increasing throughput levels. At around 45.5AC/hr the fuel consumption resulting from dynamic objective scheduling meets the minimum time solution since from here on it is necessary to solely optimize for minimum time to cope with the high required throughput.

Apart from this, comparing the dynamic objective scheduling solution to the solution obtained by the FCFS-sequence scheduling, a fuel consumption benefit of 4.8% is achieved at 30AC/hr which is slowly decreasing to about 2.2% at 42AC/hr. Higher average throughput levels of up to 48AC/hr can only be achieved by dynamic objective scheduling.

To conclude, the developed optimization tool has successfully found a balance between fuel-optimal arrival scheduling at lower traffic levels and time-optimal scheduling whenever traffic levels increase. An average fuel consumption benefit of 2.2-4.8% was achieved with respect to the currently prevailing FCFS-sequence scheduling for throughput levels. Also, a 14% higher average throughput was achieved than the FCFS-scheduling was able to cope with. Finally, it has to be mentioned that potentially more fuel-optimal scheduling solutions can be found using a more sophisticated strategy for scheduling using dynamic objectives. Especially for throughput levels between $36AC/hr$ and $42AC/hr$, the dynamic objective scheduling solution performs significantly worse than the solution that is obtained by scheduling solely for minimum fuel, which indicates that potentially more fuel benefits can be gained.

6 CONCLUSION

The combination of rapidly growing air traffic, limited airport runway capacity and the always present drive for more fuel-economic air transport raised the interest in the topic of Arrival Scheduling and Trajectory Optimization. Continuous Descent Approaches are known to significantly lower fuel consumption and noise during the descent phase but are currently only be performed during low traffic night time hours due to lacking trajectory predictability leading to larger required separation distances. Assuming Trajectory Based Operations, more time-efficient arrival sequencing and scheduling within the TMA could significantly increase the effective runway capacity, compared to the currently prevailing first-come first-served (FCFS) scheduling strategy.

This thesis project addresses an important research gap in the field of Arrival Scheduling, which is the simultaneous optimization of the arrival scheduling problem (ASP) and the aircraft trajectory optimization problem (ATOP), optimizing the corresponding approach trajectories in a dynamic optimization environment. By simulating a dynamic optimization environment a key aspect of arrival scheduling, namely the effective runway capacity must be taken into account since accumulated delay will propagate onto future aircraft arrivals. This effective runway capacity introduces an interesting trade-off between fuel-optimality and time-optimality of scheduling solutions.

The purpose of this thesis project is to balance this trade-off between fuel consumption and time-efficiency based on the presently required runway throughput aiming to achieve the most fuel-efficient arrival schedules without accumulating delay. In order to perform the task of throughput-based combined optimization of arrival schedules and approach trajectories in a dynamically changing arrival environment, an optimization tool is developed that couples fuel information of travel time based trajectories to the scheduling process by means of aircraft-specific fuel curves. Consequently the tool performs a second trajectory optimization to resolve conflicts between the trajectories corresponding to the found arrival schedule. The Freeze Horizon concept is applied together with Constrained Position Shifting to confine the scheduling problem down to a size that the optimization tool can solve in real-time.

The scope of this thesis project is limited to the last approach phase within the TMA only and will assume a single runway system. All incoming arrival aircraft will follow an identical lateral approach path with an individually optimized vertical path, applying continuous descent operations. Schiphol's night approach route, the ARTIP-3B transition, is used as the lateral approach path model for the test cases, since this route is already being used for continuous descent approaches during night time hours at Schiphol Airport. The scheduling- and trajectory optimization process is performed in the scheduling horizon, just outside the TMA-entry at around 60 miles away from the runway at flight level FL100. Once an aircraft enters the TMA, which serves as the freeze horizon, the aircraft's trajectory and therefore its place in the landing queue is fixed.

Lastly, uncertainties such as lacking trajectory predictability and variable wind speeds are not taken into account.

The designed optimization tool is able to successfully balance between fuel-optimality and time-optimality by dynamically switching between a fuel-minimal and a time-minimal scheduling objective, based on the currently required throughput level. The optimization tool applies a hybrid optimization method, solving the ASP by means of a sequential quadratic programming algorithm while solving the corresponding approach trajectories as an optimal control problem using GPOPS, a Matlab-based software which employs a class of methods that is referred to as hp-adaptive Gaussian quadrature collocation.

A throughput threshold level (TTL) is chosen to define at what required throughput level fuel-optimal scheduling must be switched to time-optimal scheduling, to cope with the higher levels of incoming traffic. Although the theoretical throughput limits were tested to be $42AC/hr$ for fuel-minimal scheduling and $48AC/hr$ for time-minimal scheduling, it is found that the TTL must be set to $36AC/hr$ in order to be able to consistently schedule arriving aircraft up to the maximum achieved average throughput limit of $48AC/hr$. Switching to time-minimal scheduling later by increasing the TTL will remove accumulated delay too late and result in infeasible landing schedules. As a reference, the prevailing FCFS-scheduling reached a limit threshold of $42AC/hr$ for the same test cases. A delay-minimal scheduling objective was considered as well, however it was found that delay is mostly caused by accumulation and propagation which is countered more effectively by the time-minimal scheduling objective. To conclude, the designed optimization tool consistently achieves a 14% higher effective runway capacity than the FCFS-scheduling. In terms of fuel consumption the optimization tool provides a 2.2-4.8% benefit over the FCFS-schedule. This benefit is shown to decrease with increasing levels of required throughput since the higher the required throughput level is, the more the optimization priority shifts towards time-optimality, minimizing wake separation distances.

Conflicts between approach trajectories, that result from the found optimal schedules, are resolved by solving the ATOP with an additional penalty function that dramatically increases the cost function of the optimal control trajectory problem whenever the minimum radar separation between two trajectories is violated. The optimization tool is able to resolve 98% of the caused conflicts, which on average results in a 3.5%-5.5% increase in fuel consumption, depending on the aircraft's weight class. It is shown that the unsolved conflicts can be resolved by increasing the initial altitude of the approach trajectory in most cases.

An important design choice for the developed optimization tool is the application of constrained position shifting. Limiting the maximum position shift (MPS) an aircraft is allowed to shift within the sequence of incoming arrival aircraft significantly lowers the amount of possible arrival sequences that need to be tested for optimality and therefore cuts valuable computation time. This report shows that setting the MPS to 3 positions is optimal, as it does not negatively affect the optimality of the scheduling solutions and is not overbearing overbearing in terms of the required computational power.

To further confine the scheduling problem, the Freeze Horizon concept is applied. Traditionally the Freeze Horizon concept defines an area-based scheduling horizon, only considering aircraft within this area in the scheduling process. However since this thesis project bases its scheduling decisions on dynamically changing levels of required throughput, it was chosen to base the scheduling horizon size on a fixed number of incoming aircraft. The scheduling horizon was chosen to contain a fixed number of 8 aircraft to be considered in each scheduling iteration. This number is shown to be sufficient to use the full scheduling potential of the optimization tool. Increasing the number of considered aircraft further results in a too large number of possible arrival sequences and cannot be handled by the available amount of memory of the used computer.

Also, scheduling based on normalized fuel curves is tested, as this maintains a sense of fairness between different weight classes of aircraft. However, when neglecting conflict resolution, this is shown to significantly increase the overall fuel consumption by 8%-12%, depending on the applied scheduling objective. Even when accounting for conflicts, which are caused less when scheduling based on normalized fuel curves, this does not compensate for the scheduling-induced fuel increase. Therefore a different solution must be found to compensate lighter weight class aircraft for their lower scheduling priority.

Lastly, as the next chapter will explain, future research could try to improve the fuel optimality of the scheduling solutions by accounting for the caused conflicts within the scheduling process and by developing a more sophisticated dynamic scheduling objective strategy. Lastly, the conflicts that were not solvable by the designed optimization tool could be solved by performing a cooperative conflict resolution as well as scheduling arrival aircraft further away from the runway. Both of these recommendation would increase the fuel-optimality of the trajectory- and scheduling solutions at the same time, either by resolving a conflict more fuel-efficiently or preventing the caused conflict entirely.

7

Recommendations & Future Work

In order to continue research on the combined topic of arrival scheduling, trajectory optimization and conflict resolution, this chapter contains recommendations on how to improve on choices made in this thesis research and on what aspects could be interesting to explore in more detail in future research. These recommendations are listed below.

- **Global Optimum of the combined ASP and ATOP.**

  In optimization one generally aims at finding the absolute best possible solution to the optimization problem, the *global optimum*. However in large complex optimization problems such as this arrival problem, where the solutions of multiple sub-problems are dependent on each other, finding the global optimum is impossible, especially since the dynamic optimization environment creates a non-deterministic situation. In this thesis research pre-computed aircraft-specific fuel curves were used to link the expected fuel consumption of the ATOP to the ASP. Although the designed optimization tool is able to find the globally optimal solution to the ASP itself by checking every possible sequence for optimality, it does so based on incomplete information. The missing information is the effect of potential conflicts to the fuel consumption. The used fuel curves assume that the approach trajectory corresponding to the computed travel time is free of conflicts. Therefore it is possible that a less fuel-optimal scheduling solution would ultimately result in a lower fuel consumption, if the corresponding approach trajectories cause less conflicts.

  The only way to find a more fuel-optimal solution to this arrival problem is by solving the ATOP for every single feasible arrival sequence, after which the best solution is chosen. Unfortunately, it is safe to say that this will be far to computationally expensive for real-time use. However, it could be interesting to explore whether more optimal scheduling results can be achieved by, for example, counting the caused conflicts of the best 5 scheduling solutions and accounting for these conflicts in the expected fuel consumption, before choosing one of these scheduling solutions.

- **Earlier Arrival Scheduling.**

  For this thesis research only the final phase of the descent within the TMA was considered, starting at about 10000ft altitude and 60NM away from the runway. This was done in order to find out whether sequencing within the TMA and thus within current approach procedure designs would provide a significant benefit in terms of fuel consumption and effective runway capacity and to check whether this is possible at all without causing unsolvable problems. As was pointed out in section 5.3.1, to prevent unsolvable conflicts it would be beneficial to sequence arrival aircraft earlier, especially at a higher altitude.

More importantly however, scheduling incoming aircraft further away from the runway would extend the interval of achievable landing times, bounded by the earliest- and latest achievable landing time, which would significantly increase scheduling optimization potential. Since this scheduling time-interval for the considered approach route in this thesis research is in the range of 320 seconds a maximum position shift of 3 positions would not always be possible. Therefore, in order to test the full potential that arrival scheduling can provide, a longer approach route has to be chosen which means that the scheduling process has to take place further away from the runway.

- **Cooperative Conflict Resolution.**

  As explained in section 4.7.3, the decision was made to solve conflicts non-cooperatively in order of the trajectory's corresponding landing times. This decision was made because the GPOPS optimal control software is not compatible with a cooperative conflict resolution which would require the amount of state variables to change within the optimization phase as conflicting aircraft feature different initial- and final times (i.e. entry times and landing times). Unfortunately non-cooperative conflict resolution is a less fuel-efficient way of solving a conflict and also leads to unnecessary situations in which conflicts cannot be solved at all while a cooperative solution would be possible.

  Fortunately this limitation to the used optimal control method is only a limitation of the GPOPS software and not a limitation of the optimization method itself. The Gaussian quadrature collocation method that GPOPS applies is theoretically able to work with variable numbers of state variables. Therefore it would be interesting for future research to see how the fuel based objective performance would change when solving conflicts cooperatively.

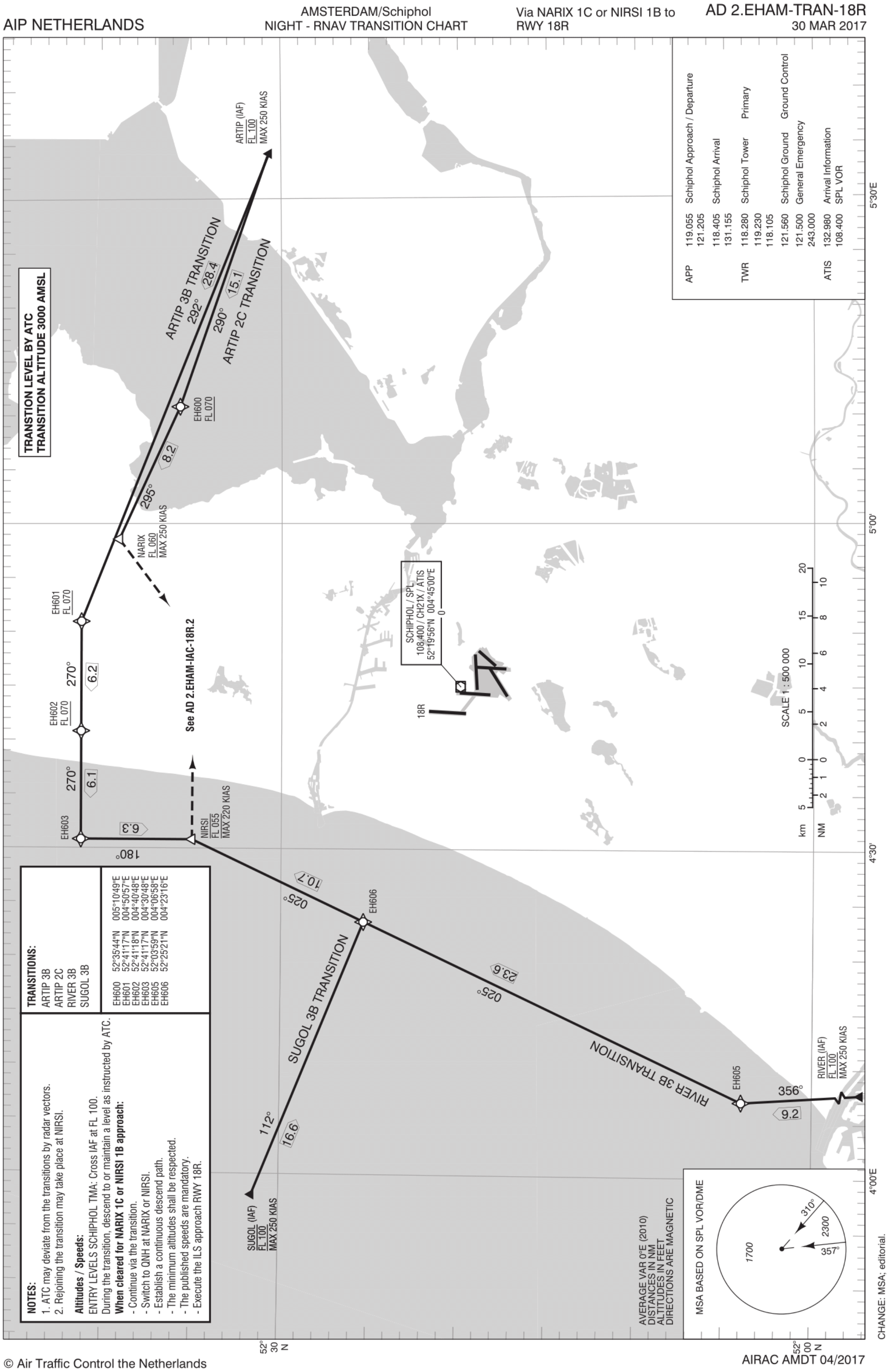
- **Dynamic Scheduling Objective Strategies.**

  Lastly, a very interesting topic to dive into for future research is the application of dynamic scheduling objective strategies, where for each scheduling iteration an optimization objective is chosen based on the currently available information of the incoming traffic stream (e.g. the required runway throughput). Although this thesis research already showed that more fuel-efficient scheduling can be achieved by using dynamic scheduling objectives, especially at higher traffic throughput levels, it can safely be assumed that a much more significant reduction in the average fuel consumption can be achieved by using a more sophisticated dynamic objective strategy. However, performing a sensitivity analysis in order to derive a more sophisticated strategy will require many more test case optimization runs than the number of runs that were performed in this thesis project, which in turn will require adequate computation power.

A

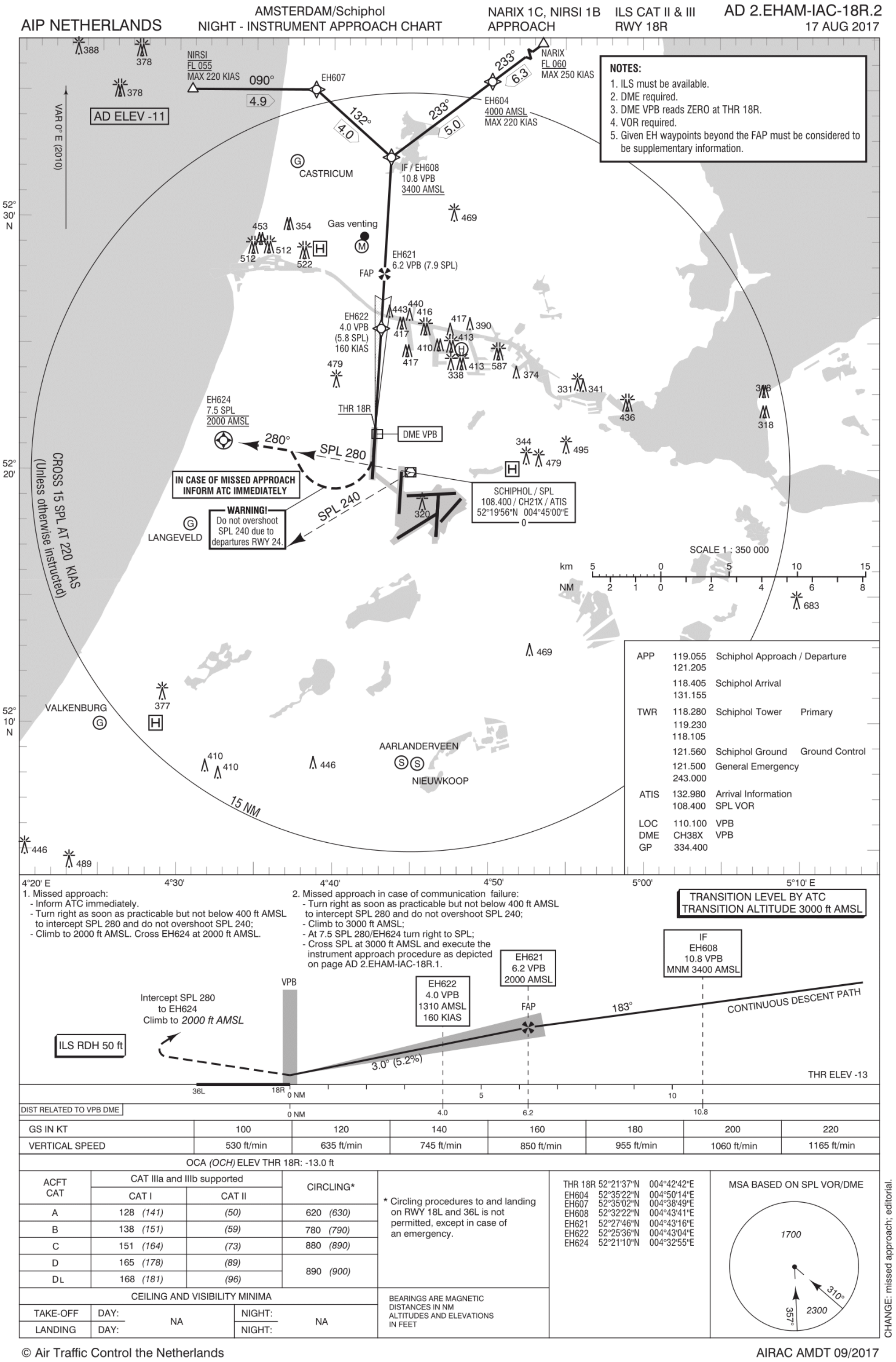
APPENDICES

A.1. Schiphol Airport - Night Instrument Approach Chart (Part 1)



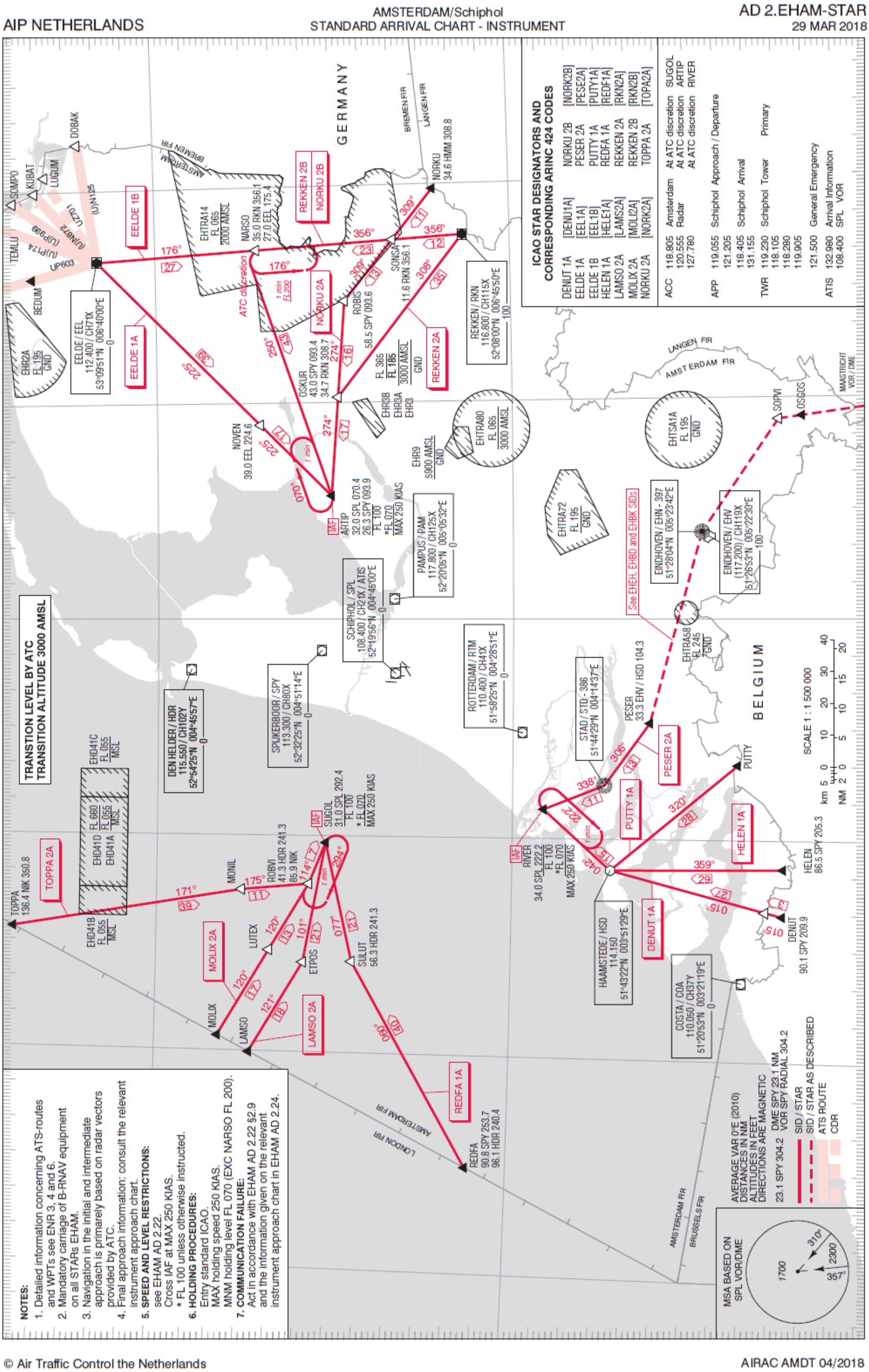
Night Instrument Approach Chart Part 1 (from AIS-Netherlands [42])

A.2. Schiphol Airport - Night Instrument Approach Chart (Part 1)



Night Instrument Approach Chart Part 2 (from AIS-Netherlands [42])

A.3. Schiphol Airport - Standard Arrival Chart



Standard Arrival Chart - Schiphol Airport (from AIS-Netherlands [42])

Bibliography

[1] S. Group, *Gebruiksprognose Amsterdam Airport Schiphol 2015* (Schiphol Group, Postbus 7501, 1118 ZG Schiphol, 2015).

[2] D. Ivanescu, C. Shaw, C. Tamvaclis, and T. Kettunen, *Models of air traffic merging techniques: evaluating performance of point merge,* in *9th AIAA Aviation Technology, Integration, and Operations Conference (ATIO) and Aircraft Noise and Emissions Reduction Symposium (ANERS)* (2009) p. 7013, uRL=https://www.researchgate.net/profile/C_Shaw2/publication/256497459_Models_of_Air_Traffic_Merging_Techniques_Evaluating_Performance_of_Point_Merge/links/02e7e5232c22262a63000000/Models-of-Air-Traffic-Merging-Techniques-Evaluating-Performance-of-Point-Merge.pdf.

[3] S. Alam, M. Nguyen, H. Abbass, C. Lokan, M. Ellejmi, and S. Kirby, *A dynamic continuous descent approach methodology for low noise and emission,* in *Digital Avionics Systems Conference (DASC), 2010 IEEE/AIAA 29th* (IEEE, 2010) pp. 1–E, uRL=https://ieeexplore.ieee.org/abstract/document/5655502/.

[4] Z.-H. Zhan, J. Zhang, Y. Li, O. Liu, S. Kwok, W. Ip, and O. Kaynak, *An efficient ant colony system based on receding horizon control for the aircraft arrival sequencing and scheduling problem,* IEEE Trantions on Intelligent Transportation Systems **11**, 399 (2010), uRL=https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5440937.

[5] F. Rooseleer and V. Treve, *Recat-eu european wake turbulence categorisation and separation minima on approach and departure,* EUROCONTROL Headquarters, Brussels (2015), uRL=https://www.eurocontrol.int/sites/default/files/content/documents/sesar/recat-eu-released-september-2015.pdf.

[6] R. G. Dear and Y. S. Sherif, *An algorithm for computer assisted sequencing and scheduling of terminal area operations,* Transportation Research Part A: General **25**, 129 (1991), uRL=https://www.sciencedirect.com/science/article/pii/019126079190132A.

[7] M. Samà, A. D'Ariano, D. Pacciarelli, K. Palagachev, and M. Gerdts, *Optimal aircraft scheduling and flight trajectory in terminal control areas,* in *5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)* (IEEE Washington, DC, 2017) pp. 285–290, uRL=https://www.researchgate.net/profile/Andrea_DAriano/publication/315754928_Optimal_Aircraft_Scheduling_and_Flight_Trajectory_in_Terminal_Control_Areas/links/58e245c94585153bfe9d9d7a/Optimal-Aircraft-Scheduling-and-Flight-Trajectory-in-Terminal-Control-Areas.pdf.

[8] L. Erkelens, *Research into new noise abatement procedures for the 21st century,* in *AIAA Guidance, Navigation, and Control Conference and Exhibit* (2000) p. 4474, uRL=https://arc.aiaa.org/doi/pdf/10.2514/6.2000-4474.

[9] J. Gibbons, *Airport System Development* (Office of Technology Assessment, Washington, D.C. 20402, 1984) uRL=https://www.princeton.edu/~ota/disk3/1984/8403/8403.PDF.

[10] T. Canada, *Aeronautical Information Manual* (Transport Canada, 2018) uRL=https://www.tc.gc.ca/media/documents/ca-publications/AIM-2018-1-E-ACCESS.pdf.

[11] J. Delsen, *Flexible arrival & departure runway allocation using mixed-integer linear programming: A schiphol airport case study,* TU Delft Repository (2016), uRL=https://repository.tudelft.nl/islandora/object/uuid:23623188-d987-49eb-883b-ea52e15f7842.

[12] B. Favennec, E. Hoffman, A. Trzmiel, F. Vergne, and K. Zeghal, *The point merge arrival flow integration technique: towards more complex environments and advanced continuous descent,* in *9th AIAA Aviation Technology, Integration, and Operations Conference (ATIO) and Aircraft Noise and Emissions Reduction Symposium (ANERS)* (2009) p. 6921, uRL=https://doi.org/10.2514/6.2009-6921.

[13] M. Ball, C. Barnhart, G. Nemhauser, and A. Odoni, *Air transportation: Irregular operations and control,* Handbooks in operations research and management science **14**, 1 (2007), uRL=https://www.sciencedirect.com/science/article/pii/S0927050706140013.

[14] J.-P. B. Clarke, N. T. Ho, L. Ren, J. A. Brown, K. R. Elmer, K. Zou, C. Hunting, D. L. McGregor, B. N. Shivashankara, K.-O. Tong, *et al.*, *Continuous descent approach: Design and flight test for louisville international airport,* Journal of Aircraft **41**, 1054 (2004), uRL=https://arc.aiaa.org/doi/abs/10.2514/1.5572.

[15] H. G. Visser, *A 4-d trajectory optimization and guidance technique for terminal area traffic management,* Delft University of Technology, Faculty of Aerospace Engineering, Report LR-769 (1994), uRL=https://repository.tudelft.nl/islandora/object/uuid:17b2cd01-8c0b-4395-aadd-1243f478b427?collection=research.

[16] Y. Matsuno, T. Tsuchiya, J. Wei, I. Hwang, and N. Matayoshi, *Stochastic optimal control for aircraft conflict resolution under wind uncertainty,* Aerospace Science and Technology **43**, 77 (2015), uRL=https://ac.els-cdn.com/S1270963815000759/1-s2.0-S1270963815000759-main.pdf?_tid=2de39cac-cc00-48fa-8f29-9a3fabfc7bdd&acdnat=1531989101_51e52b9e966f7ed5f0a417df3c857718.

[17] ICAO, *Procedures for air navigation services doc 4444,* in *Air Traffic Management 16th Edition* (999 Robert-Bourassa Boulevard, Montréal, Quebec, Canada H3C 5H7, 2016) uRL=http://flightservicebureau.org/wp-content/uploads/2017/03/ICAO-Doc4444-Pans-Atm-16thEdition-2016-OPSGROUP.pdf.

[18] G. Huijsman, *Three-dimensional conflict resolution of multiple aircraft using interior point optimization,* TU Delft Repository (2005).

[19] M. R. Jardin, *Real-time conflict-free trajectory optimization,* in *Proceedings of the fifth USA/Europe Air Traffic Management R&D Seminar* (2003) p. 85, uRL=http://www.atmseminar.org/seminarContent/seminar5/presentations/pr_027_TFO.pdf.

[20] EUROCONTROL, *Implementation guide information,* in *Continuous Descent Approach* (Postbus 7501, 1118 ZG Schiphol", 2008) uRL=https://skybrary.aero/bookshelf/books/2846.pdf.

[21] A. PUTTABAKULA, *Multi-aircraft trajectory optimization for continuous descent arrivals,* TU Delft Repository (2017), uRL=https://repository.tudelft.nl/islandora/object/uuid:99a81e51-a16a-4234-9515-c55bdf4e4a42/datastream/OBJ/download.

[22] S. G. Park, J.-P. B. Clarke, E. Feron, and H. Jimenez, *Encounter rate estimation of continuous descent arrival procedures in terminal area,* in *AIAA Guidance, Navigation, and Control Conference* (2016) p. 1630.

[23] A. P. De Leege, A. In't Veld, M. Mulder, and M. Van Paassen, *Three-degree decelerating approaches in high-density arrival streams,* Journal of Aircraft **46**, 1681 (2009), uRL=https://arc.aiaa.org/doi/abs/10.2514/1.42420.

[24] J. Klooster, K. Wichman, and O. Bleeker, *4d trajectory and time-of-arrival control to enable continuous descent arrivals,* in *AIAA Guidance, Navigation and Control Conference and Exhibit* (2008) p. 7402.

[25] K. D. Wichman, G. Carlsson, and L. G. Lindberg, *Flight trials:" runway-to-runway" required time of arrival evaluations for time-based atm environment,* in *20th DASC. 20th Digital Avionics Systems Conference (Cat. No. 01CH37219),* Vol. 2 (IEEE, 2001) pp. 7F6–1.

[26] R. Doganis, *de neufville, richard," airport systems planning"(book review),* The Town Planning Review **48**, 308 (1977).

[27] H. Erzberger, T. J. Davis, and S. Green, *Design of center-tracon automation system,* NASA Technical Reports Server (1993).

[28] H. Balakrishnan and B. Chandran, *Scheduling aircraft landings under constrained position shifting,* in *AIAA guidance, navigation, and control conference and exhibit* (2006) p. 6320.

[29] C. Morris, J. Peters, and P. Choroba, *Validation of the time based separation concept at london heathrow airport,* in *10th USA/Europe ATM R&D Seminar, Chicago*, Vol. 10 (2013) uRL=http://icrat.org/seminarContent/seminar10/papers/222-CHOROBA_0125130310-Final-Paper-4-11-13.pdf.

[30] T. Keaveney and E. Magnowska, *SESAR Solutions Catalogue 2nd Edition* (Publications Office of the European Union, SESAR JU, 2017).

[31] C. Venkatakrishnan, A. Barnett, and A. R. Odoni, *Landings at logan airport: Describing and increasing airport capacity,* Transportation Science **27**, 211 (1993), uRL=https://pubsonline.informs.org/doi/pdf/10.1287/trsc.27.3.211.

[32] A. D'Ariano, D. Pacciarelli, M. Pistelli, and M. Pranzo, *Real-time scheduling of aircraft arrivals and departures in a terminal maneuvering area,* Networks **65**, 212 (2015), uRL=https://onlinelibrary.wiley.com/doi/full/10.1002/net.21599.

[33] M. C. R. Murça and C. Müller, *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 (2015), uRL=https://www.sciencedirect.com/science/article/pii/S1366554514001938.

[34] J. Beasley, J. Sonander, and P. Havelock, *Scheduling aircraft landings at london heathrow using a population heuristic,* Journal of the operational Research Society **52**, 483 (2001), uRL=https://doi.org/10.1057/palgrave.jors.2601129.

[35] J. T. Betts, *Survey of numerical methods for trajectory optimization,* Journal of guidance, control, and dynamics **21**, 193 (1998), uRL=https://arc.aiaa.org/doi/abs/10.2514/2.4231.

[36] F. Médioni, N. Durand, and J.-M. Alliot, *Air traffic conflict resolution by genetic algorithms,* in *European Conference on Artificial Evolution* (Springer, 1995) pp. 370–383, uRL=https://link.springer.com/chapter/10.1007/3-540-61108-8_51.

[37] J. V. Hansen, *Genetic search methods in air traffic control,* Computers & Operations Research **31**, 445 (2004), uRL=https://www.sciencedirect.com/science/article/pii/S0305054802002289.

[38] K. Artiouchine, P. Baptiste, and C. Dürr, *Runway sequencing with holding patterns,* European Journal of Operational Research **189**, 1254 (2008), uRL=https://www.sciencedirect.com/science/article/pii/S0377221707005978.

[39] D. Toratani, D. Delahaye, S. Ueno, and T. Higuchi, *Merging optimization method with multiple entry points for extended terminal maneuvering area,* in *EIWAC 2015, 4th ENRI International Workshop on ATM/CNS* (2015) uRL=https://hal-enac.archives-ouvertes.fr/hal-01288387/document.

[40] A. Nuic, D. Poles, and V. Mouillet, *Bada: An advanced aircraft performance model for present and future atm systems,* International journal of adaptive control and signal processing **24**, 850 (2010), uRL=https://www.eurocontrol.int/sites/default/files/2019-03/overview-bada-apm.pdf.

[41] D. T. Lyons, *Improved aircraft cruise by periodic control.* NASA (1981), uRL=https://www.researchgate.net/publication/35200624_IMPROVED_AIRCRAFT_CRUISE_BY_PERIODIC_CONTROL.

[42] Air Traffic Control the Netherlands, *Standard arrival chart - schiphol airport,* (2018), air Traffic Control the Netherlands, URL=http://www.ais-netherlands.nl/aim/2018-07-05-AIRAC/html/index-en-GB.html, Accessed: 2018-07-18.