Master of Science Thesis

4D Trajectory De-Confliction For Future ATM By Applying Constraints

R.A.S. Baboeram Panday
1170643

Delft University of Technology

Challenge the future

4D Trajectory De-Confliction For Future ATM By Applying Constraints

THESIS

submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE

in

ELECTRICAL ENGINEERING

by

R.A.S. Baboeram Panday
born in Nw. Nickerie, Suriname

Minor Avionics
Department of Telecommunications
Faculty of Electrical Engineering, Mathematics and Computer Science (EEMCS)
Delft University of Technology

4D Trajectory De-Confliction For Future ATM By Applying Constraints

by R.A.S. Baboeram Panday

Abstract

In the present ATC system conflicts between aircraft are detected and resolved in a stated-based approach. To meet traffic demand predictions, the global air traffic management (ATM) system needs revolutionary changes. Several visions on future ATM operations exist (e.g. NextGen, SESAR). A common function between the different visions is 4D Trajectory management. This function enables trajectory-based operation as opposed to the state-based approach of the present system. Instead of monitoring the current traffic situation and resolving short term conflicts by vectoring, the 4D trajectory management function de-conflicts all trajectories prior to execution. A key function of 4D trajectory management is the resolution of 4D trajectories. Conflict resolution of 4D trajectories is applied to conflict scenarios using constraints. A characteristic of a constraint is that it does not limit the aircraft to a particular solution but provides the aircraft the room to generate a trajectory within the actual solution space. Conflicting trajectories are resolved using constraints where the constraints provides an approximation of the solution space. This concept of trajectory de-confliction through the use of constraints is the topic of this thesis work.

Track	:	Telecommunications
Student Number	:	1170643
Defense date	:	26 August, 2013
Committee Members	:	

Chairperson:	dr.ir. E. Theunissen, Delft University of Technology
Member:	dr.ir. G.J.M Janssen, Delft University of Technology
Member:	ing. O.F. Bleeker, Rockwell Collins
Member:	ir. G.J.M. Koeners, Delft University of Technology

To my father, may he rest in peace

Contents

List of Figures

List of Tables

Acknowledgements

At the start of my study at the Delft University of Technology I was always interested in Aerospace Engineering. When I started to enter the Master phase of my study in Electrical Engineering I got acquainted with the Avionics group within the track Telecommunications. The minor Avionics provided me the opportunity to explore Aerospace Engineering and my enthusiasm grew when I attended lectures like Flight Dynamics at the faculty of Aerospace Engineering. This interest in Avionics resulted into a Master's thesis in the field of 4D trajectory management where constraints are applied for conflict resolution of 4DT trajectories.

Before getting into more detail in the field of 4D trajectory management in ATM, I would like to thank Erik Theunissen as the devoted power behind the Avionics group. Erik provided me the opportunity to fulfill my Master's degree within the Avionics group. Furthermore, I would like to thank Joris Koeners who have guided me throughout this thesis work. I will not forget our conversations about this thesis as on many other things. I am convinced that both Erik and Joris have made me a better engineer.

I also would like to thank my family who have unconditionally supported me even in some difficult times. Finally, let me thank my friend and high school companion Motta, for the improvements, comments and overall support.

R.A.S. Baboeram Panday
Delft, The Netherlands
26 August 2013

1 Introduction

Ever since man first saw a bird, man wanted to fly. In December of 1903, the Wright brothers became the first people to fly a plane. Air travel has improved tremendously since the first successful attempt made by the Wright brothers. In 2008, more than 10 million flights were handled over Europe which is on average about 28.000 flights daily. In the 'most-likely' scenario of the long term forecast, there will be 16.9 million IFR movements in Europe in 2030. The range of the forecast scenarios is between 13.1 and 20.9 million flights in 2030. The growth will average 1.6%-3.9% annually (2.8% in the 'most-likely') [7] [8]. Revolutionary changes are required to accommodate the increase of air traffic and make optimal use of the airspace capacity. Several visions on future air traffic management exist like NextGen [9] and SESAR [10]. A common function between the different visions is 4D trajectory management. The 4D trajectory management function, which is responsible for conflict detection and resolution of 4D trajectories using constraints, is the subject of this thesis.
This introduction chapter is organized as follows: Section 1.1 presents background information about how conflicts are detected and handled today and in the future perspective. Section 1.2 provides the motivation for this work. In Section 1.3 project goals are identified and Section 1.4 presents the achievements. Section 1.5 concludes this Introduction chapter with an overview of this thesis' organization.

1.1 Background

Currently, as aircraft travels through a given airspace, it is monitored by one or more air traffic controllers responsible for that airspace. The controller monitor the aircraft and give instructions to ensure safe separation among other aircraft. As the aircraft leaves an airspace and enters another, the air traffic controller passes it off to the controller responsible for the new airspace. In the present ATC system conflicts are predicted using aircraft position, straight line extrapolation of the current velocity vector and intent information like a flight plan. This approach has generally some uncertainty in the estimate of the future position of the aircraft. Predicted conflicts are resolved using vectors which are provided by the air traffic controller. This approach, where the current position of the aircraft is known and the future position of the aircraft is estimated based on the velocity vector and flight plan information, is also referred as a state-based approach.

In this current framework, the airspace capacity is limited by the cognitive capacity of human controllers to maintain safe separation with high reliability [11]. Today's controller can handle approximately fifteen aircraft. Studies have found that air traffic could be at least doubled or tripled over the current limits without saturating the actual capacity of the airspace itself [12][13][14]. In the present ATC system predicted conflicts are handled by applying delays in the form of vectoring, holding patterns and speed restrictions. At areas where the air traffic is low this method operates properly. In congested areas where the traffic is high, the state-of-the-art air navigation system is running at maximum and revolutionary changes are required to accommodate the increase of future air traffic [7][15].

In the past current and future aircraft position were *estimated*, today current aircraft position is *known* and future aircraft position is *estimated*. In the prospected future, the emphasis lies on trajectory-based operations where current and future aircraft position are *known*. The increase of available information necessitate this paradigm shift which is illustrated in Figure 1.1 [16]. Instead of waiting for aircraft appearance on the radar screen and subsequently performing conflict detection and resolution, trajectory information available from the FMS can be used a priori for conflict detection and resolution.

Future Air Traffic Control is heading towards a system referred as Air Traffic Management where the emphasis is shifted from active to passive control of aircraft (or from control to monitor of aircraft) with a policy of intervention by exception [4][17]. In this framework, the aircraft is able to fly user preferred 4-Dimensional trajectories (three spatial dimension plus time). In the vision of Air Traffic Management, the aircraft position in space and time is captured in a 4-Dimensional trajectory. Studies, like the one performed by Wichman et al. [18] indicates that current state-of-the-art Flight Management Systems are capable of flying a 4D trajectory accurately. The availability of planned 4D trajectories and the ability to follow it accurately enables trajectory-based operations, i.e. conflict detection and resolution can be performed using *known* current and future aircraft position.

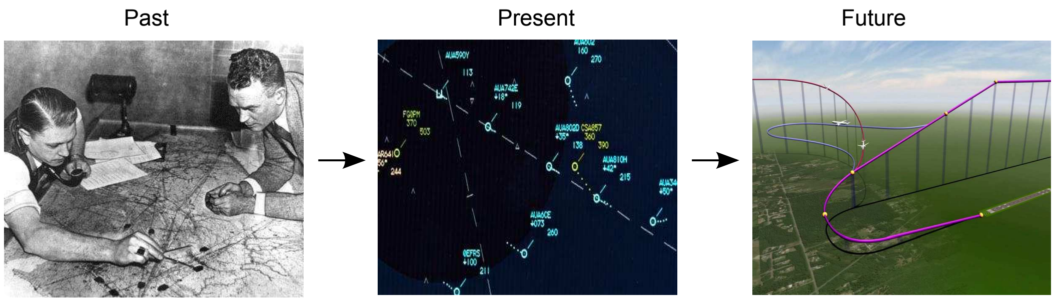

Figure 1.1: Evolution of air traffic control, past *(left)*[1], today *(center)*[2], prospected future *(right)*[3]

1.2 Motivation and scope

1.2.1 4D trajectory management

In the current centralized ATC world user preferred routes are not prime considerations. Conflicts are predicted using known aircraft position and velocity vector (state-based). This approach has generally some uncertainty in the estimate of the future position of the aircraft. As mentioned in the previous section the current way of handling air traffic air traffic is reaching its limits and in order to meet traffic demand predictions changes are required. Several visions already exists like NextGen and SESAR. A common function between the different visions is 4D trajectory management where 4D trajectories are used for conflict detection and resolution. This trajectory-based operation is envisioned to play a key role in future Air Traffic Management [9][10].

A characteristic of 4D trajectory management is that the future aircraft position in time is known and captured in a 4D trajectory. This information of the future aircraft position in time can be used to perform conflict detection and resolution a priori. Furthermore, when aircraft are able to execute a trajectory accurately, more aircraft can be accommodated in the same airspace and therefore increasing airspace capacity. This trajectory-based operation is made possible by FMS capabilities of generating and following a 4D trajectory accurately [19][18][4].

In the 4D navigation world where each user has a 4D trajectory, in which the future spatial position and corresponding time at the position is encapsulated, conflicting trajectories will inevitably arise which must be resolved in order to ensure safe operations. The challenge is how to resolve these conflicting trajectories and will be discussed in the remainder of this thesis document.

1.2.2 Scope

In trajectory-based operations, where each user has a 4D trajectory, the 4D trajectory has to be fed into the system (Flight Management System) where it can be executed. Generation of the 4D trajectory can be done by the aircraft itself or by ground based ATC. At the aircraft side aircraft parameters like amount of fuel, weight, aircraft type and user performance parameters like acquiring a time driven or economical flight are available. It is therefore likely to delegate the task of trajectory generation to the aircraft. Furthermore, a 4D trajectory generated externally of the aircraft (by ATC) can not be given as input to the Flight Management System and therefore can not be executed.

Given the above rationale, trajectory generation should thus be performed by the aircraft. Then what about conflict detection and resolution? In the state-based approach conflict detection and resolution is performed externally of the aircraft, i.e. by the air traffic controller on the ground. Within 4D trajectory management conflict detection and resolution can be achieved in different ways. Conflict detection could be performed by the aircraft itself. This requires full topology of aircraft 4D trajectories at the aircraft side in order to determine conflicts with other aircraft trajectories. From the operational point of view conflict detection should be delegated to the ground (ATC). The rationale behind this is the excess of computational resources, storage facilities (databases needed for storage of active 4D trajectories) and the availability of full topology of trajectories on the ground.

In this approach a data-link is then required to communicate the aircraft generated 4D trajectory to ATC. At the ATC side conflict detection and resolution can be performed given other aircraft trajectories. Conflict resolution can be performed by generating a revised 4D trajectory and submitting it back to the aircraft. Given the above rationale that the aircraft is best able to determine a 4D trajectory, it is not logical to resolve a conflict by providing a revised 4D trajectory as solution. Furthermore, providing a 4D trajectory as solution has an imposing character since the aircraft would be restricted to follow that particular trajectory within the solution space. Additional operational issue is that a 4D trajectory can not be accepted as input to state-of-the-art Flight Management System.

Another way to perform conflict resolution is by applying constraints to the aircraft upon detection of conflicting 4D trajectories. Applying constraints provide the user the ability to generate an user preferred trajectory while taking the imposed constraint(s) into account [4]. This way of managing 4D trajectories generate a policy of intervention by exception where ATM activities are partially delegated to the flight deck (trajectory generation) while conflict detection and resolution (constraint generation) are performed by ATM. Constraining aircraft in a minimum way would result in near-optimal trajectories [20]. This approach of resolving conflicted trajectories has already been raised by researchers like Wichman et al. [4] and is elaborated on in Chapter 3. The part of 4D trajectory management where constraints are used for conflict resolution of 4D trajectories is the subject of this thesis work.

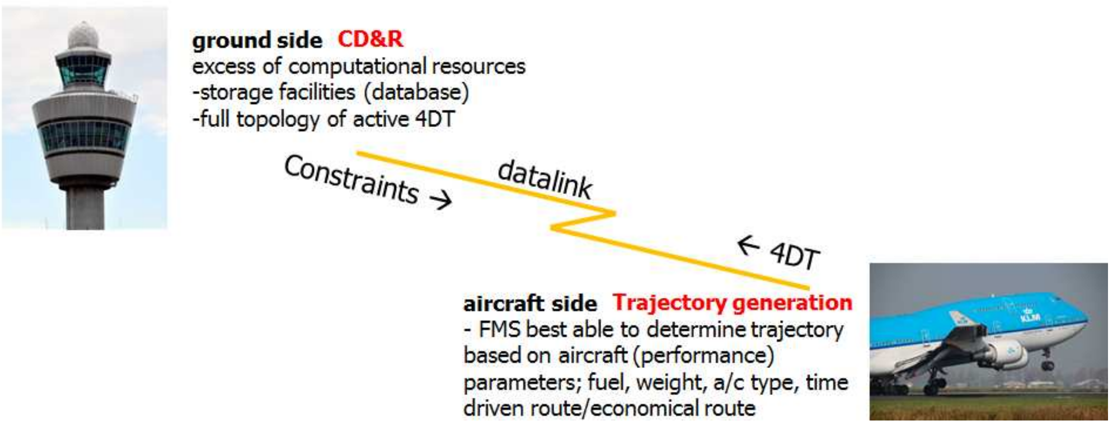

Figure 1.2: Aircraft is delegated with the task of trajectory generation whereas at the ATM side conflict detection and resolution is performed.

1.3 Thesis goals

The goal of this research assignment is to implement and evaluate a 4D de-confliction algorithm where constraints are used to resolve conflicts. To reach the goal, the assignment is divided in the following stages:

1. Perform literature survey into existing (4D) trajectory de-confliction algorithms.
2. Classify the algorithms and identify the issues. Define rating criteria to compare the different algorithms.
3. Choose a concept, design and implement in either Matlab, C or C++.
4. Evaluate the implementation using the defined rating criteria.
5. Discuss the results.

1.4 Achievements

A literature study is performed into existing 4D de-confliction algorithms. From the literature study follows that the focus mainly lies on resolution algorithms where the solution consists of a trajectory rather than of constraints. Using a trajectory as solution to a conflict, restricts, besides the drawbacks discussed in 1.2.2, the aircraft to that particular trajectory within the solution space.

The performed literature study did not show an algorithm that provides constraints for conflict resolution which resolved a conflict. The focus in the literature study shifted into the search of constraints which could be used to resolve conflicting 4D trajectories. The search for constraint is performed by looking at; how can a conflict be resolved (lateral, vertical, in time), what the ideal constraint would be, limitations in state-of-the-art technology to apply this ideal constraint, messages over existing data-link and by looking at the input capabilities of a state-of-the-art Flight Management System. Constraints are identified and extracted from a FMCS guide which seems promising for resolving conflicting 4D trajectories [21].

The possible identified constraints found in the literature study were designed and strategies were derived to determine the appropriate constraint values. In the vision of using constraints in this work, the constraint provides an approximation of the solution space. This approximation is achieved by defining ranges for the identified constraints in the later, vertical and time domain. Since this approximation of the solution space (using constraints) lies within the actual solution space any constraint value within the range is expected to resolve the conflict.

In order to obtain the constraint values (the upper and lower bound which defines the range) design strategies are derived. In a case study the identified constraints and design strategies are applied to conflict scenarios obtained from the NASA research database [22]. The conflict scenarios involved two aircraft in conflict. An intruder aircraft is added to obtain the upper bound. The result of this first implementation (resolutions of the conflict scenarios) provides constraints for all scenarios on which the approach is applied on, even in the case other aircraft was assigned to be the maneuver aircraft. Preliminary conclusions can be made that the constraints derived can be used to approximate the solution space ensuring that a constraint value within the provided range resolves the conflict.

1.5 Thesis organization

This thesis is organized as follows. First, in Chapter 2 the outcome of the literature study regarding the topic of conflict resolution of 4D trajectories is presented. In this chapter among others the constraints, which are expected to be applicable for conflict resolution according to the vision in this work, are identified. In Chapter 3 the Concept of Operation in which constraints are used for trajectory de-confliction is presented. Furthermore in Chapter 3 the identified possible constraints are discussed in more detail and methods to determine the constraint values that resolves the conflict are presented. The identified constraints and approaches to determine the constraint values are applied to conflict scenarios. In Chapter 4 a conflict scenario is used to illustrate the approach and vision of conflict resolution through the use of constraints. Chapter 5 discusses the applied approach for conflict resolution of 4D trajectories in this work and recommendations for future work are provided. Chapter 6 concludes this thesis work with a summary and conclusion.

2 Literature study

In order to accommodate predicted air traffic increase, revolutionary changes are required. One of the changes is the shift towards trajectory-based operations (TBO) where 4D trajectories are used for conflict detection and resolution. In this approach, aircraft have a 4D trajectory which it is able to follow accurately [19][4][23]. The availability of 4D trajectories requires functions like conflict detection and resolution to resolve detected conflicting 4D trajectories. Detected conflicts between aircraft 4D trajectories could be resolved by applying constraints to the aircraft as described in the previous Chapter 1. A literature study, in the field of 4D trajectory management, where the emphasis lies on de-confliction of 4D trajectories through the use of constraints, has been performed and is presented in this chapter.
This chapter is outlined as follows: first in Section 2.1 terminology frequently used in the field of 4D trajectory management is discussed. Subsequently the goal of the literature study is given Section 2.2. In Section 2.3 conflict resolution in the state-of-the-art in the field is discussed. In Section 2.4 the derivation and identification of constraints which can be used for conflict resolution is discussed. Subsequently in Section 2.5 methods to determine these constraints values are discussed. Finally, Section 2.6 ends this Chapter with an evaluation of the literature study.

2.1 Terminology

Before discussing the results of the literature study, it is noteworthy to provide the definition of terminology frequently used in the field of 4D trajectory management and in this thesis document. The following definitions are adopted in the remainder of this work.

4D trajectory

The definition of a *4D trajectory* according to the SESAR Consortium (2007) is given as [24]:

> *A set of consecutive segments linking waypoints and/or points computed by FMS (airborne) or by TP (ground) to build the vertical profile and the lateral transitions; each point defined by a longitude, a latitude, a level and a time.*

The 4D trajectory contains more specified information about the future aircraft position in space and time than a flight plan which is currently used to estimate future aircraft position. A flight plan does not provide the aircraft position in space and time between the waypoints which builds the flight plan.

A conflict

The term *conflict* defines an event in which two or more aircraft experience a loss of minimum separation, i.e. the distance between aircraft violates criteria defining what is undesirable [25]. The minimum separation criteria are typically 5 nmi horizontal and 1000 ft vertical. This can be seen as a protected zone (PZ) surrounding an aircraft that should not be penetrated by another aircraft (see Figure 2.1 and Figure 2.2).

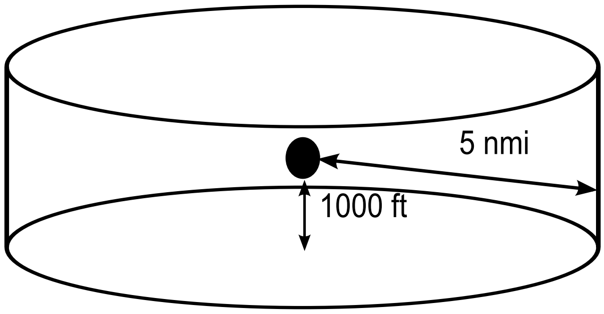

Figure 2.1: Aircraft protected zone, the black dot represents the aircraft in the center of its PZ. Another aircraft may not enter this PZ.

In Figure 2.2 parts of two aircraft trajectories are depicted. A conflict occurs when at a common time instance the spatial separation requirement is violated. In this case the aircraft have the same altitude and the minimum separation between the spatial positions is required to be at least 5 nautical miles. From Figure 2.2 it can be seen that the both trajectories are predicted to be in conflict between time instances t_{33}-t_{36} should it continue along these trajectories. Either the horizontal separation requirement (5 nautical miles) and/or the vertical separation requirement (1000 feet) should be met in order to avoid a conflict.

Solution space

The solution space is the complex 4-dimensional space in which an aircraft trajectory is free to be in within and has no conflict with other aircraft trajectories, i.e. the minimum horizontal and vertical separation distance (3D) with own aircraft and other aircraft is not violated at any time.

Constraint

A constraint limits the aircraft in space and or time in which it can generate a trajectory.

2.2 Goal of the literature study

Given the 4D trajectories of aircraft, the goal of the literature study was to find algorithms or approaches that resolves conflicting 4D trajectories using constraints. The functionality which is looked for, in the state-of-the-art in the field related to 4D trajectory management, is illustrated in Figure 2.3 where the red box represents conflict resolution algorithms or approaches which when provided with conflicting 4D trajectories generate constraints ensuring that the conflict is resolved upon conforming to the constraints.

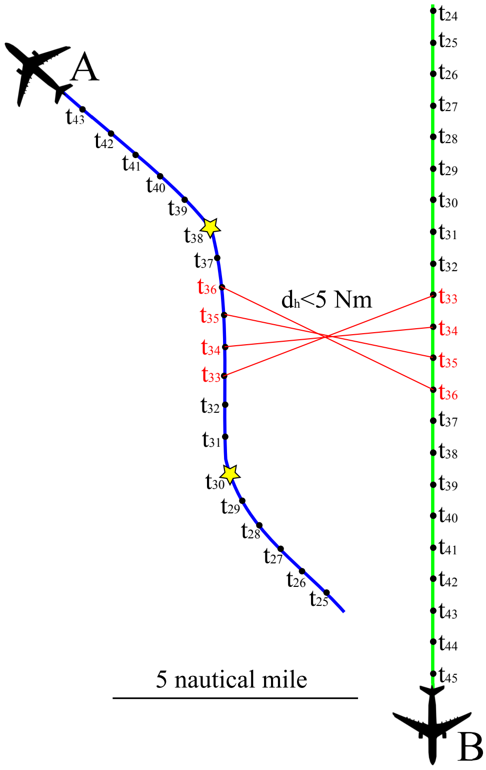

Figure 2.2: Illustration of a horizontal conflict between two aircraft trajectories at a common altitude. Aircraft A is heading northwest and trajectory Aircraft B south. Operating at the same altitude does not imply that the aircraft are in conflict unless the horizontal separation requirement is violated, which is the case here and illustrated by the red time instances.

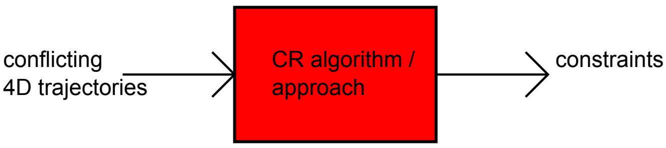

Figure 2.3: Illustration of the initial focus in the literature study, i.e. algorithms or approaches that provide constraints to resolve conflicting 4D trajectories.

2.3 Conflict resolution in state-of-the-art in the field

As stated in Section 2.2 the goal of the literature study was to find conflict resolution algorithms that generate constraints to resolve conflicting 4D trajectories. Different approaches regarding conflict detection and resolution are found in the literature [14][26][27][28][25][23][29][30][31][32][33][34][35][36]. The literature study did not show any algorithm or approach that resolved conflicting 4D trajectories using constraints. Typically, research related to 4D trajectory management (conflict detection and resolution) involved resolutions which are provided in the form of trajectories or parts of trajectories [26][27][28][23][30][31]. These trajectories may not always be the best solution as another trajectory (which is different in the route, speed profile or both) within the solution space may be preferred by the user which also results in conflict free situation.

In other research, the emphasis relies on state-based conflict resolution where the current position and velocity vector is used to estimate future aircraft position after which conflict detection and resolution is performed [32][33][34][35][36].

A common similarity between the approaches found in the literature is the imposing character of the conflict resolution, i.e. the solution trajectory is imposed on the aircraft providing the aircraft no freedom to determine its own trajectory within the solution space.

The vision is to use constraints for conflict resolution according to the rationale stated in Section 1.2.2. This is indicated in Figure 2.3 and thus it is desirable that a conflict between aircraft trajectories is resolved by using constraints. In Figure 2.4 this concept of conflict resolution through the use of constraints depicted. The concept of using constraints to resolve conflicting trajectories has already been raised by researchers like Wichman et al. [4][19]. It would be desirable that the conflict is resolved directly after imposing constraints in the first cycle, i.e. upon detection of a conflict constraints are generated which when imposed guarantees that the conflict is resolved. The iteration cycle of trajectory generation by aircraft and conflict detection and resolution by ATM is then limited to one cycle. In the literature no algorithm is found which generates appropriate constraints to resolve conflicting trajectories nor are the constraints indicated which could be used in this approach.

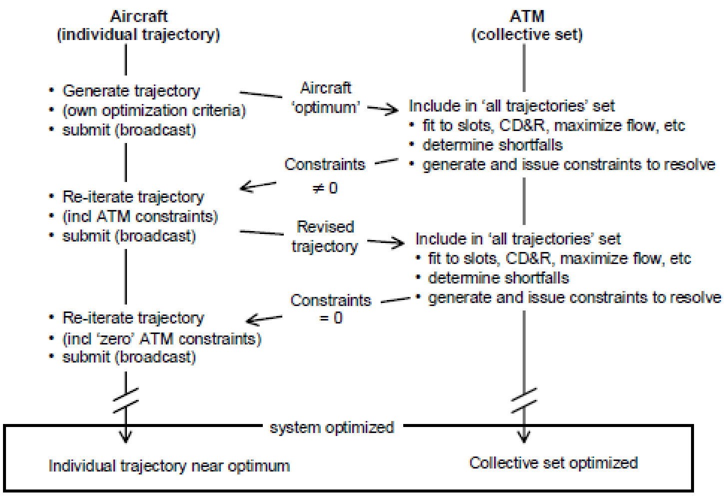

Figure 2.4: Conflict resolution with trajectory-constraints negotiation [4]

2.4 Identification of possible constraints

Due to the lack of algorithms that have the functionality which is searched for in the literature and illustrated in Figure 2.3, the focus in the literature study shifted from *algorithms* which should provide constraints to *constraints* itself. Particularly constraints, which when imposed on aircraft and ensuring that the conflicts are resolved, were of interest.

Given a conflict, the aircraft trajectories involved generate a conflict region in space and time. The conflict region is the overlapping volume of two cylinders for every common time instance. The center of the cylinder corresponds to the aircraft position and the dimension of the cylinder is defined by a radius of 2.5 nmi and height of 1000 ft. Overlapping cylinders at common time instances generates a conflict region. In the ideal case the conflict region in space and time could be seen as a constraint and the on-board FMS could be provided with this information forcing the aircraft to generate a 4D trajectory outside the conflict region. Current state-of-the-art FMS is not capable of accepting the complex conflict area in space and time as input and subsequently generate a trajectory outside this region. Therefore the approach of determining the conflict region in space and time, subsequently imposing this as a constraint to the aircraft seems not feasible.

The inverse approach is to provide the solution space to the aircraft (and not the region in space and time which an aircraft is not allowed to be within). Constraints could be used to provide the user this information, i.e. the solution space, in which it is allowed to generate a 4D trajectory in space and time. According to this vision, the constraints of interest thus should approximate the complex 4-Dimensional (space and time) solution space.

Upon encountering a conflict an aircraft can typically resolve the conflict laterally in the xy plane, vertically by acquiring another altitude or in time by slowing down or speeding up [37]. Given these three domains a search is made for constraints which could approximate the solution space in these domains. An example of a constraint in the time domain, which is used nowadays by the air traffic controller, is the Required Time of Arrival (RTA) constraint typically applied at a waypoint for regulating traffic flow and maintaining safe separation. This RTA constraint restricts the aircraft in the time domain whereby a conflict with a dynamic obstacle can be resolved. The RTA constraint will be discussed later on in this section.

2.4.1 Messages over existing data-link

Since data-link is required, as has been discussed earlier in Section 1.2.2, it is a matter of course to analyze messages which are communicated over existing data-link. The reason for this is to find out if constraints, in the domains described above, could be identified and extracted from existing data-link messages.

In the literature, the Controller Pilot Data Link Communications (CPDLC) message set has been found which provides a means of communication between controller and pilot using data link for ATC communication. The controller is provided with the capability to issue level assignments, crossing constraints, lateral deviations, route changes and clearances, speed assignments, radio frequency assignments, and various requests for information. The pilot is provided with the capability to respond to messages, to request clearances and information, to report information, and to declare an emergency. In Europe, the CPDLC service is available in the Maastricht Upper Area Control Centre where connection is initiated after the aircraft is airborne and at least 30 minutes before entering the Maastricht Upper Area Control airspace. For aircraft departing from an airport located close to the airspace managed by the Maastricht Upper Area Control Centre, the CPDLC connection will only occur above FL 150, to minimize crew distraction [38].

The messages in Table 2.1 are identified from the CPDLC message set which restricts the aircraft in space and or in time. The messages where a range is specified are of interest because they could approximate the solution space in a certain domain. For example, the message element # 52 specifies a range in time whereas message element # 51 restricts the aircraft to one solution in time and thus not provides the aircraft the range, i.e. the solution space, in which it is allowed to operate. If the solution space consists of just one solution in time, of course the message element #51 is the appropriate message to use. In the time domain message elements #52, #53, #54 and in the vertical domain the message elements #47, #48 and #50 provides the ability to specify ranges.

Msg#	Message element
46	CROSS [position] AT [altitude]
47	CROSS [position] AT OR ABOVE [altitude]
48	CROSS [position] AT OR BELOW [altitude]
49	CROSS [position] AT AND MAINTAIN [altitude]
50	CROSS [position] BETWEEN [altitude] AND [altitude]
51	CROSS [position] AT [time]
52	CROSS [position] AT OR BEFORE [time]
53	CROSS [position] AT OR AFTER [time]
54	CROSS [position] BETWEEN [time] AND [time]
63	AT [time] CROSS [position] AT AND MAINTAIN [altitude] AT [speed]
65	AT [position] OFFSET [distance offset] [direction] OF ROUTE

Table 2.1: CPDLC messages [6]

2.4.2 FMS input capabilities

ATC is responsible for conflict detection and generates constraints in order to resolve conflicting trajectories. This vision has been discussed earlier in Section 1.2.2. In this vision constraints are transmitted over data-link and ultimately fed in to the system at the aircraft side. In the previous subsection, the data-link capabilities were discussed. Next, a Flight Management System guide was analyzed to determine the FMS input capabilities and to see if constraints can be identified in the different domains (lateral, vertical and time).

In the literature, only the Smiths FMCS guide was found that provided a comprehensive overview of the functionality of a Flight Management Computer [21]. The Smiths

FMCS guide provides operational information of the Flight Management Computer of a Boeing 737-600/700/800. Other manuals were typically used for flight simulation software and provided similar functions of the FMC. However, they did not provide a full overview of the FMC functionality like the Smiths FMCS guide[39][40][41]. Therefore, the Smiths FMCS guide is analyzed and used as a basis for determining FMS input capabilities and subsequently identifying possible constraints.

The following input capabilities are derived from the Smiths FMCS guide [21] and categorized according to the resolution domain as described above. In the search for possible constraints, constraints which could approximate the solution space, i.e. constraints which have the ability to specify ranges and not restricting the aircraft to one particular solution, were of interest. The following FMS input capabilities, which seems feasible in this approach, are identified and their characteristic according to the Smiths FMCS guide are discussed [21].

- The **Along Track Offset (ATO)** is applied at a waypoint. A positive value of the ATO corresponds with a forward shift of the waypoint, whereas a negative value corresponds to a backward shift of the waypoint in the trajectory. The ATO constraint can be seen as restricting the aircraft to fly a different route laterally.

- **Lateral Offset (LO)** is applied to a leg and is constructed by specifying the starting and end waypoint of the leg and an offset distance. A positive offset distance corresponds to an offset to the right and a negative offset corresponds to an offset to the left. As in the case of an ATO, the LO can be seen as restricting the aircraft to fly a different route laterally.

- **Altitude (ALT)** In the vertical domain altitude restrictions can be applied to a waypoint. Altitude or Flight Levels followed by an "A" are "at or above" targets, a "B" indicates an "at or below" target, and no letter suffix means the altitude is an "at" target to be met. Two altitude values indicate an altitude "window" or block crossing restriction for that waypoint. The ALT constraint can be seen as restricting the aircraft to fly a different route vertically.

- The **Required Time of Arrival (RTA)** advisory and control capabilities provide a performance speed schedule to arrive at a specific point at a specific time. The Required Time of Arrival is applied at a specified waypoint with "at", "at or before", "at or after" targets to be met.

- **Speed (SP)** entries can be applied to climb, cruise and descent legs. In climb legs, the speed constraint is a restriction for that leg only. In cruise legs, the speed entries are continued on to the Top-of-Descent (ToD) from the leg the entry is made. In the descent leg, the speed entries are continued on down path until a leg with a lower speed is encountered. A common characteristic of Speed entries and the Required Time of Arrival is that they only affect the timing profile of the 4D trajectory.

2.4.3 Possible constraints

In the ideal situation the complex conflict region in space and time could be provided as constraint to the aircraft. Because of operational limitations of state-of the-art FMS to accept this conflict region as input and subsequently generate a 4D trajectory outside this region, a search is made to state-of-the-art possibilities to resolve conflicting 4D trajectories according to the vision stated in Section 1.2.2. In this approach existing data-link capabilities were investigated and a state-of-the-art FMS is analyzed in order to find possible constraints. The vision is to use constraints to approximate the solution space as stated earlier in this Section and in Section 1.2.2. In this approach it is required that the constraints specifies a range in order to inform the user the solution space in which it can generate a trajectory upon detection of a conflict.

From the existing CPDLC message set and FMSC guide possible constraints are identified, which are discussed in Sections 2.4.1 and 2.4.2 respectively. The messages identified from the CPDLC message set, which seems feasible for conflict resolution, are comparable with the input capabilities found in the Smiths FMCS guide, i.e. message elements #47, #48, #50 and #52, #53, #54 with the Altitude and Required Time of Arrival respectively. Because of operational limitations of the CPDLC service (see Section 2.4.1, comparable functionality of the possible identified constraints and expected application to state-of-the-art FMS (since the derivation of possible constraints is from a state-of-the-art FMS), the identified constraints from the FMCS guide are used in this work. Some of the possible identified constraints does not provides a range which is required, like the lateral offset and along track offset, but are promising to be applicable for conflict resolution in the envisioned approach, i.e. using constraints to approximate the solution space. This is further elaborated on in Chapter 3 where the possible identified constraints from the Smiths FMCS guide are discussed in more detail.

2.5 Methods to determine the constraint values

As already stated earlier, the conflict region cannot be provided as a constraint to state-of-the-art FMS. The possible constraints obtained from the Smiths FMCS guide seems promising in the sense that they could provide an approximation of the solution space. Imposing the constraints can thus be seen as a way to inform the aircraft of the solution space (by specifying ranges) within it is allowed to generate a trajectory. The question arises how to determine the appropriate constraint values, i.e. what are the ranges of the offsets (ATO, LO, ALT) or targets (RTA, SP)?

The emphasis in the literature study shifted again, this time to algorithms or functions which could generate the appropriate constraint values. The constraints values specifies the range of a particular constraint type and therefore could be used to approximate the solution space.

The algorithms and functions found in the literature, which are of interest in this approach, are described next in Section 2.5.1, Section 2.5.2, Section 2.5.3 and Section 2.5.4. Table 2.2 provides an overview of them and with their corresponding characteristics. In the columns of the table, the algorithm characteristics like input, output and solution domain are listed. References to the papers are provided for more detailed information about the algorithms.

2.5.1 Heading change conflict resolution algorithm

The paper of Bach et al. [42] documents a simple level turn algorithm. The paper presents the derivation of the heading which results in a just miss of the protected zone of the intruder aircraft given the known aircraft positions and constant ground velocity vectors. The method uses a relative conflict geometry where the intruder aircraft is stationary with respect to the relative maneuver aircraft. In this approach, the aircraft velocities are kept unchanged. One can say that this is a state-based approach of conflict resolution where state information (current position and velocity vectors) are used for determining a heading of the maneuver aircraft which satisfies the minimum separation requirement[1]. The obtained heading for the maneuver aircraft, which resolves the conflict, cannot be read in as an input to the FMS. However, this algorithm seems promising of converting the new heading change, which results in conflict free situation, into a lateral offset. This will be further elaborated on in Chapter 3.

2.5.2 Speed change conflict resolution

Koeners [5] presented a separation assessment tool that supports the air traffic controller with his task of conflict resolution. The tool uses an analytical method to compute the speed range that would result in a loss of separation.

Given the required separation and the speed of one of the aircraft, the Closest Point of Approach[2] (CPA) is computed as a function of trial speeds. The speed range corresponding to the CPA's with a separation greater than the minimum requirement are feasible speed values. There are three possible outcomes of the speed (range) resulting in no loss of separation:

1. no solution is found, given the speed of one aircraft no speed of the other (maneuver) aircraft result in a CPA meeting the required separation,

2. only one speed is found which result in exactly the required separation,

3. there are two speed values at which the CPA is exactly the minimum required separation. All speed between those values will result in CPA with a separation below the required minimum separation.

The speed (range) can be used to calculate a RTA constraint. This is also elaborated on in Chapter 3.

2.5.3 Time-based conflict resolution

Idris et al. [37] developed a conflict resolution algorithm that resolved predicted conflicts by time shifting (delaying or advancing) one of the flights prior to conflict. The conflict

[1] The minimum required separation distance is typically 5 nmi horizontally and 1000 ft vertically.

[2] The Closest Point of Approach of an aircraft is a point along the aircraft trajectory at which the distance between the aircraft is at its minimum value.

resolution algorithm was applied to conflicts involving a horizontal intruder aircraft with constant ground speed and constant altitude and a descending maneuver aircraft. However, the approach is generic and could be applied to more general conflict scenarios. The method uses relative geometry of the conflict situation to derive the conflict region in the time-space domain from which the time shift can be determined. Upon absorbing the (positive or negative) time shift the conflict region is avoided in time. The time shift cannot directly be accepted as input to state-of-the-art FMS but can be used to derive a RTA constraint and is discussed further in Chapter 3.

2.5.4 Conflict resolution by ground delay

Barnier and Allignol [43] discusses 4D trajectory de-confliction through departure time adjustments. A CP model is used to determine the ground delay by which predicted conflicts between aircraft trajectories, occurring above a given flight level, are resolved. The approach can be seen as an aircraft scheduling problem where ground holding is used to obtain a conflict free time slot. However, for a predicted conflict, a conflict free slot may be found which result in an unacceptable delay for the airliner or a conflict free slot may not be found at all. This approach of determining a ground delay could be used to acquire a time shift constraint (RTA or speed as described in Section 3.4).

2.6 Evaluation of the literature study

The initial goal of the literature study was to find algorithms in the literature that resolved conflicting 4D trajectories by providing constraints. The literature study in state-of-the-art in the field did not showed any algorithm that meets this functionality. The focus in the literature was mainly on conflict resolution where a 4D trajectory is provided as solution [26][27][28][30]. This approach is not in line with the vision of providing a solution space instead of imposing a single solution within the solution space to the aircraft as stated in Section 1.2.2.

A search is made to constraints which could be used to approximate the solution space. Possible constraints are identified by looking at existing data-link capabilities and by analyzing a FMS guide. The possible constraints, requirements and issues are discussed. Furthermore, algorithms or functions to determine the appropriate values of the identified possible constraints are discussed. The next step is to explore if the identified constraints and methods to obtain the appropriate values are feasible for conflict resolution of 4D trajectories. This is discussed in the upcoming Chapters.

	Input	Output	Domain	Operation phase(s)	Remarks
Heading change algorithm [42]	current position and velocity vector	new heading	lateral	cruise, can be used for all phases by projecting cf in hor. plane	promising for generation of lateral offset constraint
Speed range [5]	current position and velocity vector	feasible speed range	time domain (speed solution)	cruise	promising for generation of RTA
Time-based conflict resolution [37]	trajectory	time delay	time (only the maneuver aircraft absorbs a pos/neg time shift prior to the predicted cf)	considered to descent phase, could be applicable to climb and cruise	approach can be used to determine RTA constraint
CR by ground delay [43]	trajectory	ground delay in the form a free time slot	time, the whole 4D trajectory is shifted in time	n/a	slots may not be found or the amount of ground delay may be too large

Table 2.2: Overview of the characteristics of algorithms that generate a constraint values or are promising to determine the constraint value of the possible constraints identified from the Smiths FMCS guide.

Case Study 3

The literature study showed that research in the field of 4D trajectory resolution mainly relies on resolving a conflict by imposing a trajectory as solution rather than applying constraints. The lack of an algorithm that generates constraints led to the search of constraints itself. Constraints which seems applicable in this vision and methods to determine the constraints values are discussed in the previous Chapter. This Chapter discusses a case study in which the identified possible constraints and methods to determine the constraint values are applied to conflict scenarios. The design of the constraints and methods to determine the constriants values are discussed in this Chapter.
First, in Section 3.1 a short introduction is given. In Section 3.2 the Concept of Operation is discussed and the vision of how constraints are used for resolving conflicting trajectories is explained. Subsequently, the conflict detection function, which is used in the case study, is discussed in Section 3.3. The identified constraints and methods to determine the constraints values are discussed in Section 3.4. Finally, Section 3.5 discusses the application of the approach on conflict scenarios.

3.1 Introduction

The vision is to use constraints to resolve conflicting trajectories. Particularly, constraints which could approximate the solution space are of interest. Using constraints in this approach can be seen as notifying to the aircraft the solution space in which it is allowed to re-generate a trajectory after a conflict has been detected on the ground. In the previous Chapter possible constraints are identified which could be used for conflict resolution. The constraints, together with their issues and requirements in order to be applicable in this approach are discussed in more detail in this Chapter. Furthermore, the methods to determine the appropriate constraint value are discussed. A first case study is performed in order to explore if the derived constraints and corresponding methods are feasible for conflict resolution of 4D trajectories. In the case study the constraints and corresponding methods are applied to conflict scenarios by implementing the system in MATLAB R2013a.

3.2 Concept of Operation

In the vision where constraints are used for conflict resolution of 4D trajectories, the aircraft is delegated with the task of trajectory generation whereas ATM is responsible for conflict detection and resolution. The rationale behind this has already been discussed in Section 1.2.2 and illustrated in Figure 1.2. The aircraft generates a preferred trajectory and submits this to Air Traffic Management. At the ATM side the collective set of trajectories is known and conflict detection is performed. Upon detection of a conflict constraints are generated and imposed on the aircraft, otherwise the requested trajectory is approved and ready to be executed.

The constraints provide an approximation of the solution space in which the aircraft can generate a revised trajectory. This process is illustrated in Figure 3.1 and differs from the process in Figure 2.4 in the sense that there is no negotiation and the imposed constraints is guaranteed to resolve the conflict. This seems obvious since, according to the applied vision of conflict resolution, the constraints provide a approximation of the solution space (where the approximation of the solution space is a subset of the actual total solution space).

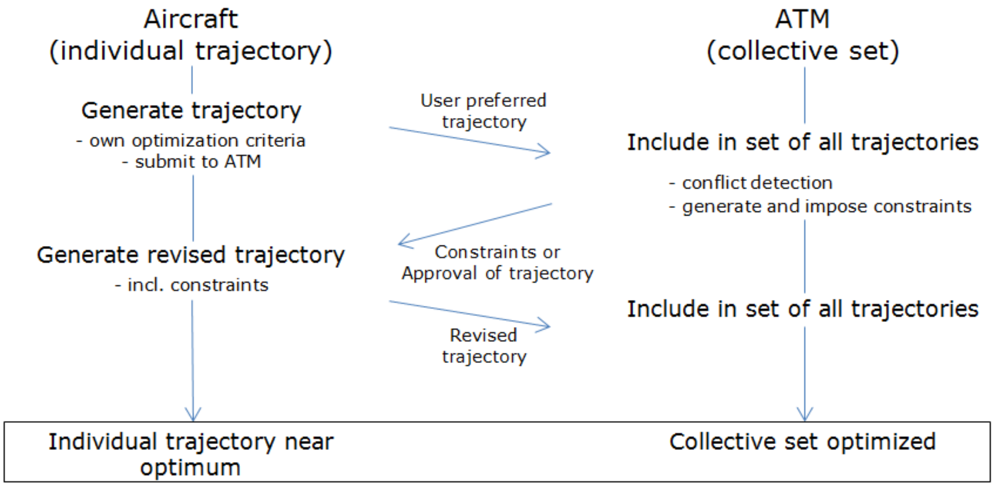

Figure 3.1: Conflict detection and resolution process

According to this approach the trajectory is made available at ATM side and thus a data-link is required for communication of the trajectory and constraints. To justify the omission of the negotiation process, as illustrated in Figure 2.4, it is required that the approximation of the solution space (through the use of constraints) lies within the actual total solution space.

Feasible constraints generated by ATM are submitted to the aircraft from which the aircraft can determine the most appropriate solution (e.g. apply a constraint which results in the most economical trajectory or trajectory with the smallest time delay).

A flowchart is used in the design process of the described system above. It helps to visualize what is going on and to understand the process. The flowchart, function blocks and position with respect to the resolution process depicted in Figure 3.1 are illustrated in Figure 3.2.

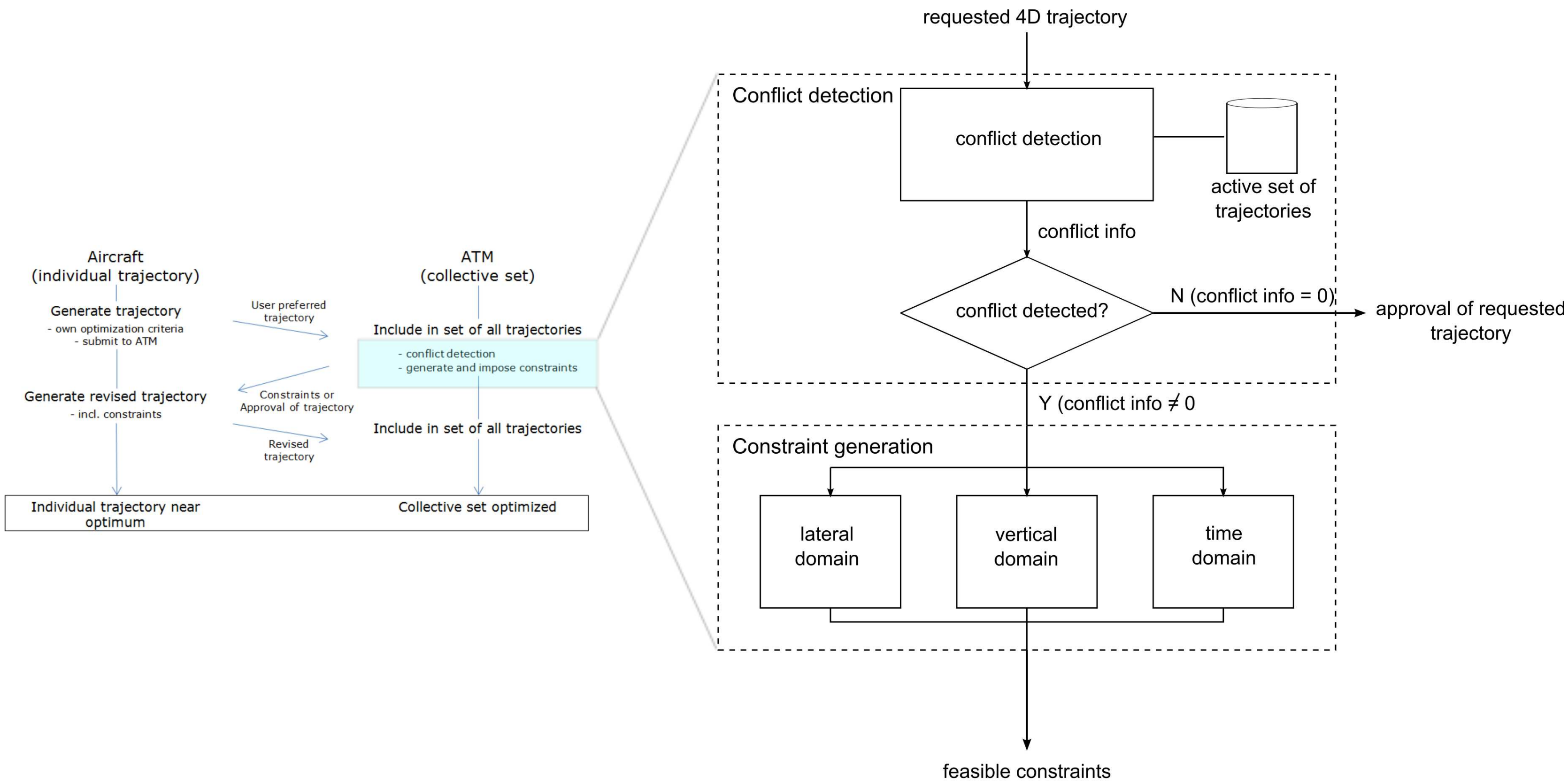

Figure 3.2: Flowchart and position w.r.t. resolution process

The input to the system is the requested (user preferred) 4D trajectory and the output are constraints in case a conflict is detected. If no conflict is detected, the requested trajectory is approved and ready to be executed. Upon detection of a conflict, constraints are derived in the three domains: lateral, vertical and time domain. In order to ensure that the generated constraint resolves the conflict, i.e. results in that the aircraft generates a revised trajectory which is free of conflict, the constraints have to provide an approximation of the solution space where this approximation lies within the total solution space.

A requirement of a constraint is that it is able to specify a range. This range is required in order to provide the solution space to the aircraft (through the use of constraints). If the constraints only would specify one solution instead of a range of solutions, it would restrict the aircraft to that particular solution and would mask other solutions. The term constraint would therefore not apply. A constraint which is used currently by the air traffic controller is the Required Time of Arrival constraint. Applying for example an RTA ”at or before” target by ATC forces the aircraft to arrive earlier in time at a certain point on which the RTA constraint is applied. The RTA provides a range in time at which the aircraft is allowed to be at a certain point in space, and thus a range in speed which the aircraft can acquire in order to meet the target.

Given a conflict with two aircraft trajectories, the range of a constraint would typically be defined by a lower bound and an ”infinite” bound. Referring back to the RTA ”at or before” constraint, the lower bound would be the time instance T. Furthermore any time instance before T would also be fine (the ”infinite” bound). Given the RTA constraint, the on-board FMS can determine the feasible speed range which would result in meeting the target. In practice, applying an RTA constraint with only a lower bound could result in a secondary conflict with for example another aircraft in which is in trail. In this case it is then required to specify the range by an upper and lower bound in which the aircraft is free to acquire a target without inducing a secondary aircraft.

As already mentioned, the system accepts a 4D trajectory as input and conflict detection is performed. After a conflict is detected, constraints are generated. The conflict detection function and constraint generation function are discussed next in the upcoming Section 3.3 and Section 3.4.

3.3 Conflict detection

Input to the system, as illustrated in Figure 3.2, are 4D trajectories. In the literature study conflict scenarios and corresponding 4D trajectories were found on the NASA website [22]. The conflict scenarios include the 4D trajectories of two conflicting aircraft. Instead of manually generating 4D trajectories, which are in conflict, conflict scenarios generated by an air traffic simulation program developed by NASA are used [22]. The conflict data available on the NASA website were used for a conflict resolution algorithm discussed by Farley and Erzberger [28]. This approach provides resolution where the solution typically is a trajectory, which is not desirable as discussed in the previous Chapter. However the conflict scenarios and corresponding 4D trajectories of the aircraft could be used to demonstrate the concept of conflict resolution through the use of constraints.

3.3.1 Format of the 4D trajectory

The definition of a 4D trajectory according to the SESAR consortium (see Section 2.1) does not indicate exactly how the 4DT is built up. The trajectories obtained from the the NASA database have a time step of 10 seconds, i.e. for every 10 seconds the 3D spatial position (latitude, longitude, altitude) is captured. The trajectories could only be obtained by copying it from the website and pasting it in for example a .txt or .xls file. This was a time consuming effort since space errors had to be corrected (for typically about 150 records) before it could be read into Matlab.

However, the NASA database also contained the list of waypoints corresponding to the trajectories. Using the list of waypoints (typically 15-20 records) and applying extrapolation between the waypoints, the same trajectories were obtained with minimal discrepancies (only at the waypoints). In the extrapolation process time steps of 1 second were chosen because the time at the waypoints were not expressed as a multiple of 10 seconds. A function has been created in Matlab which reads the NASA trajectory information (list of waypoints) and subsequently generates a trajectory with time step of 1 second.

Defining a 4D trajectory with time step with for example 10 seconds would reduce the amount of miss detections then in the case a 4D trajectory is defined by a set of waypoints. In the last case, there is a higher uncertainty about the aircraft position in time between the waypoints (or the assumption or agreement should be made that the speed between the waypoints is constant). For the illustration of the concept of trajectory de-confliction through the use of constraints discussed in this case study, it didn't matter if trajectories with time step of 1 second or 10 seconds were used. For simplicity and ease of use a time step of 1 second is chosen. According to Barnier and Allignol a maximum time step of 15 seconds is allowed [43].

Drawbacks of this approach, where a trajectory is defined with time step of 1 second, are storage and computational time. Alternatively, parts of the 4D trajectory where the aircraft maintains a constant speed (for example in the cruise phase of flight) may be omitted in the 4D trajectory. Agreements should then be made about constant speed in these parts of the trajectory. Another option would be to include the speed profile for every waypoint, e.g. by using an analytical way to define the 4D trajectory. At the ground side, the whole trajectory can them be constructed. Another possibility would be to only capture the spatial deviation from the previous aircraft position in trajectory for every time step. Again at the ground side the whole 4D trajectory can then be constructed. These options depend on how the FMS provides the trajectory as output.

In the vision of this work the on-board FMS provides more detailed information of future aircraft position in time than just a list of waypoints (where there is uncertainty of the aircraft position in time between the waypoints). This is required in order to perform conflict detection and prevent missed detections at the ATM side. The way in which the trajectory information could be encapsulated is mentioned in the above. At the ATM side, the whole trajectory can then be constructed for an agreed time step (maximum of 15 seconds according to Barnier and Allignol [43]). The constructed trajectory can then be used for conflict detection and resolution. In this work the assumption is made that the FMS provides a 4D trajectory which is specified by waypoints and time at the waypoints and constant speed between the waypoints. At the ground side a 4D trajectory is then constructed with a time step of 1 second which is then used to perform conflict detection and resolution.

3.3.2 Conflict detection

Given above definition of the 4D trajectory adopted in this case study, conflict detection had to be performed in order to determine where along the trajectories a conflict is predicted. The conflict detection block accepts as input the requested 4D trajectories of the aircraft. As output conflict information is given which consist of the indices of the trajectories where a conflict is predicted. The part of the trajectory which is in conflict is in the remainder termed as the conflicting points.

The spatial 3D position *(x,y,z)* and time at these position build a 4D trajectory. The time step is chosen to be 1 second, i.e. for every second the 3D position of the aircraft is captured in the 4D trajectory.

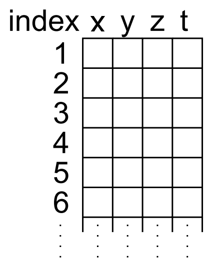

Figure 3.3: Format of a 4D trajectory

Conflict detection is performed by checking whether the separation norm is violated for every common time instance. Since the trajectories may not have the same length or start and end at the same time, the common time needs to be determined and for these common time instance the separation norm is tested.

The horizontal distance d_h and vertical distance d_v between two aircraft (A and B) positions at common time instance is defined by

$$d_h = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2} \tag{3.1}$$

$$d_v = |z_A - z_B| \tag{3.2}$$

The separation norm is then defined by

$$d_h > 5Nm \quad || \quad d_v > 1000ft \qquad \forall t \in t_A \cap t_B \tag{3.3}$$

The separation norm requires that the horizontal distance should be larger than 5 nautical miles and or the vertical distance should be larger than 1000 feet for all common time instances. In Figure 3.4 two scenarios are depicted which illustrates a conflict situation between two aircraft at the same altitude in the left half of the figure and a conflict free situation in the right half of the figure. Due to another time schedule

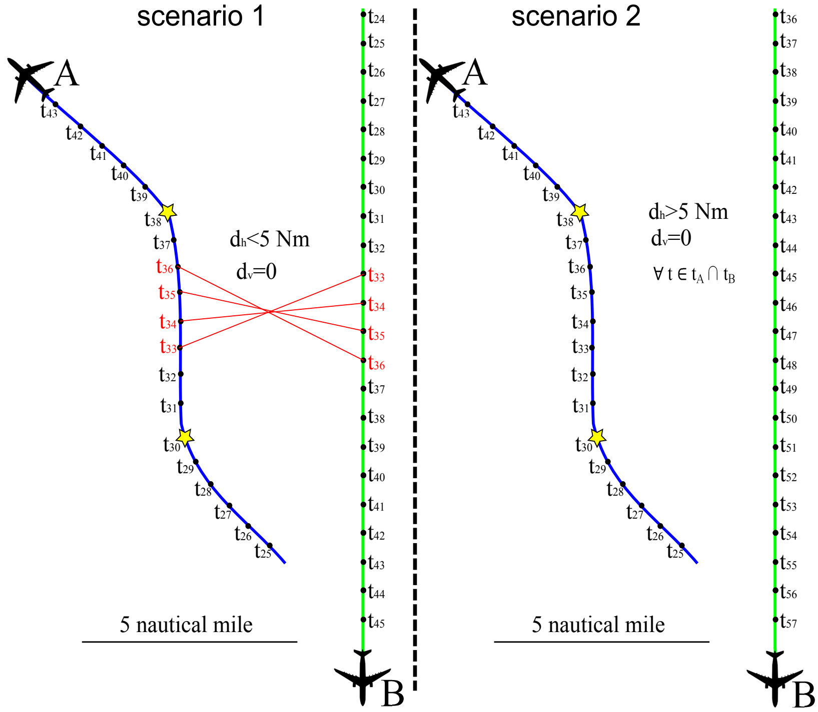

Figure 3.4: In the left situation a conflict is detected where both the horizontal and vertical separation requirement are violated. In the right situation, only the vertical separation requirement is violated whereas the horizontal requirement is met, ensuring that the aircraft do not enter each other protected zone.

in scenario 2 the aircraft do not enter each other protected zone and are therefore not in conflict.

This conflict detection function is implemented in Matlab which determines the conflicting points by returning the indices of the conflicting part of the trajectories.

3.4 Constraint generation

Given the conflicting part of a trajectory which is determined in the previous Section 3.3, the next step is to determine constraints which resolves the conflict. In Section 2.4.3 possible constraints were identified which could be applied according to the approach discussed in the Concept of Operation. This Section discusses these possible constraints in more detail together with the corresponding approach to determine the constraint value.

3.4.1 Constraints in the lateral domain

In the lateral domain, the along track offset (ATO) and lateral offset (LO) were identified as possible constraints. The along track offset and lateral offset provide a way to alter the aircraft trajectory in the horizontal plane. By applying an along track offset, a waypoint is shifted forward or backwards (depending on whether a negative or positive ATO is applied). In the case of a lateral offset, a leg is offset negatively or positively (to the left or to the right respectively).

As already mentioned earlier in Section 2.4.3 the lateral offset and along track offset does not provide a range like in the case of an RTA ”at or before” for example. By small adjustments to the lateral offset and along track offset, they could be converted to constraints that specify a range in which the aircraft is allowed to operate. This is achieved by defining a upper and lower bound for the feasible range. In this way the solution space in the lateral domain is approximated using the lateral offset or along track offset constraint.

If the negotiation process would be used (Figure 2.4) a range would not be required. In the trajectory negotiation process, the aircraft has the possibility to apply an absolute ATO or LO value greater than an ATO or LO value which is imposed on the aircraft. The aircraft subsequently generates a revised trajectory and submits this back to ATM for another cycle of conflict detection and resolution untill a feasible ATO or LO is found. In the vision as defined in the Concept of Operation, this is not desirable or needed when using the solution space approach, i.e. the solution space is provided to the aircraft by translating (approximating) it into (using) constraints.

In the remainder of this Section the way the appropriate along track offset and lateral offset range is determined are discussed.

Along track offset constraint

In the literature study, no method or algorithm was found which provided an along track offset value. Therefore, a method is derived that provides the appropriate value for the along track offset constraint which ensures that the conflict between two conflicting 4D trajectories is resolved.

According to the description of the along track offset derived from to the Smiths FMCS guide, waypoints in the trajectory could be shifted positively or negatively along track [21]. This shifting of a waypoint is illustrated in Figure 3.5 using a fictitious and simple scenario. The Figure displays a top view of an trajectory with the part of the trajectory that is in conflict (Figure 3.5a). In Figure 3.5b waypoint B has been offset negatively. The feasible range for the an along track offset is defined by the minimum (lower bound) and maximum (upper bound) offset. Upon imposing an along track offset constraint, the aircraft can apply any offset within the specified range which results that the conflict is resolved. Similarly, applying a positive along track offset constraint at waypoint C is illustrated in Figure 3.5c.

The feasibility of the along track offset constraint strongly depends on the conflict geometry. Applying an along track offset constraint to a waypoint which is between two consecutive legs having the same heading make no sense since the aircraft will follow the same spatial route. Therefore, along track offset constraint should be applied to waypoints which are between two consecutive legs that have different heading. Applying a ATO constraint can be seen as imposing the aircraft to acquire another trajectory within a specified range (the solution space) in the horizontal plane. The first and last waypoint can not be offset since they correspond to the departure and destination airport respectively.

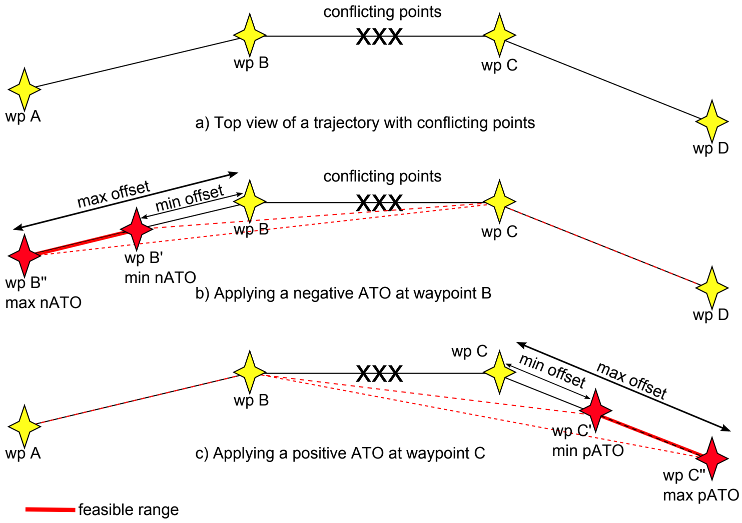

Figure 3.5: Top view of a trajectory where the concept of using ATO is illustrated.

Provided with the above characteristics about the along track offset constraint, the following strategy is used to determine the offset (upper and lower bound) which ensures that the conflict is resolved. Given the conflicting points obtained from the conflict detection function, the first step is to determine which waypoint will be constrained with an ATO. In case of an negative ATO, the first waypoint prior to the conflicting points is chosen as waypoint to be constrained. This waypoint is negatively offset by shifting the waypoint backwards along the trajectory. The opposite strategy is used for the first waypoint after the conflicting points in case a positive ATO constraint is applied. In case waypoint lies within the conflicting point, this waypoint is chosen as waypoint on which the along track offset constraint is applied on.

The maximum offset can be as long as the leg length of the leg prior to the waypoint on which the offset is applied (for a negative offset constraint) or as long as the leg length of the leg after the waypoint on which the offset is applied (for a positive offset constraint) [21]. In practice, the maximum offset can be smaller than the leg length due to other active aircraft trajectories. As mentioned before, it is required that the waypoint which is subject to an along track offset constraint lies between two consecutive legs having different headings. If this in not the case, no feasible ATO constraint is found.

In pseudo code, the strategy for determining the along track offset (negative and positive) is written as:

$$\begin{gathered}
\textbf{negative ATO:} \\
\textit{determine wp on which ATO is applied} = \textit{wp}\#\mathrm{x} \\
\downarrow \\
\textit{wp}\#\mathrm{x} = \textit{wp}\#1 \ ? \xrightarrow{Y} \textit{no feasible neg ATO} \\
\downarrow N \\
\textit{heading check:} \\
heading(leg_{\#x-1}) = heading(leg_{\#x}) \ ? \xrightarrow{Y} \textit{no feasible neg ATO}
\end{gathered}$$

$$\downarrow N$$
$$\textit{negatively offset wp}\#\text{x}$$
$$\textit{max offset} = \textit{leglength}(\textit{leg}_{\#x-1})$$

positive ATO:

$$\textit{determine wp on which ATO is applied} = \textit{wp}\#\text{x}$$
$$\downarrow$$
$$\textit{wp}\#\text{x} = \textit{last wp ?} \xrightarrow{Y} \textit{no feasible pos ATO}$$
$$\downarrow N$$
$$\textit{heading check:}$$
$$\textit{heading}(\textit{leg}_{\#x-1}) = \textit{heading}(\textit{leg}_{\#x}) \textit{ ?} \xrightarrow{Y} \textit{no feasible pos ATO}$$
$$\downarrow N$$
$$\textit{positively offset wp}\#\text{x}$$
$$\textit{max offset} = \textit{leglength}(\textit{leg}_{\#x})$$

The strategy described above stepwise offsets the waypoint for a negative and or positive along track offset constraint. For every offset step, the corresponding trajectory is determined. Applying only an along track offset constraint is not sufficient since the aircraft may comply to the along track offset constraint (thus spatially comply to shifting the waypoint within the specified range) but still may construct a speed (or time) profile of the 4D trajectory which ultimately still may incur the conflict. Therefore, it is also required to restrict the conflicting part of the aircraft 4D trajectory in the time domain. In the approach chosen here for determining the along track offset constraint, the aircraft speed in the legs of the trajectory is considered to be fixed. In other words, upon detection of a conflict the aircraft is provided with an along track offset constraint which specifies the range within a waypoint should be offset. By ATM it is then guaranteed that the conflict is resolved as long as the speed in the legs of the revised trajectory (which is generated by the aircraft taking the imposed constraint into account) remains the same as the speed in the legs of the original trajectory.

Lateral offset constraint

As in the case of the along track offset, no method is found in the literature study which directly provides the lateral offset constraint (LO). The level turn algorithm described by Bach et al. is used to derive the lateral offset constraint [42].

According to the description of the lateral offset as stated in the Smiths FMCS guide [21], legs can be offset laterally. The lateral offset is specified by the starting and end point of the leg and the corresponding offset distance. A positive offset corresponds to an offset to the right whereas a negative offset corresponds to an offset to the left of the to be offset leg. As in the case of the along track offset, a range has to be specified in order to consider the lateral offset as constraint (see the rationale in Section 2.4 and at the beginning of this Section). The lateral offset constraint is illustrated in Figure 3.6 using a fictitious trajectory. The minimum offset (lower bound) and maximum offset (upper bound) which specifies the feasible range (which can be seen as the solution space) of the lateral offset are illustrated. Considering a conflict with two aircraft trajectories, the solution space of the lateral offset constraint would be specified by a lower bound and an

”infinite” upper bound. In practice, an upper bound can be needed to avoid secondary conflicts with other aircraft.

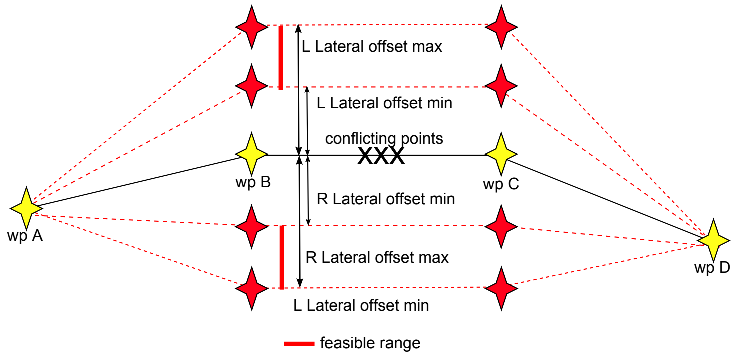

Figure 3.6: Top view of a trajectory where the concept of using LO is illustrated.

Provided with this description of the lateral offset constraint, a strategy needs to be determined which provides the appropriate offset ensuring that the conflict is resolved. The approach of Bach et al. is used which provides the heading which results in just miss of the protected zone of the other aircraft involved. Appendix C describes the derivation of the heading as discussed by Bach et al. [42]. Given the new heading ψ_1^* obtained from the level turn algorithm and the distance from the begin point of the leg up to the last conflicting point in the leg ds, the offset distance is given by

$$LO = ds \sin \Delta\psi \tag{3.4}$$

where $\Delta\psi$ is the heading difference between the original planned heading of the leg and the heading obtained from the level turn algorithm

$$\Delta\psi = \psi_1^* - \psi_1 \tag{3.5}$$

The lateral offset is applied to the conflicting leg of the trajectory, i.e. the leg where the conflicting points lies within. This approach requires that the aircraft speed profile is kept unchanged, i.e. the speed in the legs is the same as in the requested 4D trajectory. Furthermore, the level turn algorithm requires that the speed of both aircraft involved is constant. If these requirements are not met, the derivation of the offset as described above is not feasible. The lateral offset constraint is then determined in a stepwise manner as in the case of the along track offset constraint. The leg to be offset is then stepwise offset (either positive or negative) until the minimum required offset is found. Another possibility would be to constrain the aircraft to maintain a certain constant speed in the leg involved and subsequently derive the heading from the level turn algorithm pretending that the speed is constant (which is required by the level turn algorithm). The offset is then determined using equations 3.4 and 3.5.

In the approach of determining the lateral offset constraint in this case study the lateral offset constraint is only applied when the conflicting points lies in one leg. Offsetting multiple neighboring legs, e.g. when the conflicting points are ”spread” over multiple legs, requires a way to connect the offset legs with each other. A method to do this would be to find the crossing point of the offset legs. This crossing point can be seen as an auxiliary waypoint which connects the to be offset legs. Furthermore, the offsets of the legs have to be in the same direction (to the left or right of the original trajectory). For simplicity, only one leg can be offset and the conflict points are required to be in the leg which is to be offset.

In pseudo code, the strategy for determining the lateral offset (negative or positive) is:

$$\begin{gathered}
\textbf{lateral offset (constant speed case):} \\
\textit{conflicting points lies in a leg ?} \xrightarrow{N} \textit{no feasible LO} \\
\downarrow Y \\
\textit{determine conflicting leg} = leg_{\#\mathrm{x}} \\
\downarrow \\
\textit{determine heading from level turn algorithm} \\
\downarrow \\
LO = ds \sin \Delta\psi \\
\downarrow \\
\textit{lateral offset} = LO
\end{gathered}$$

$$\begin{gathered}
\textbf{lateral offset (not constant speed case):} \\
\textit{conflicting points lies in a leg ?} \xrightarrow{N} \textit{no feasible LO} \\
\downarrow Y \\
\textit{determine conflicting leg} = leg_{\#\mathrm{x}} \\
\downarrow \\
\textit{determine average speed of } leg_{\#\mathrm{x}} = v_{av_{leg\#x}} \\
\downarrow \\
\textit{construct trajectory, replace } leg_{\#\mathrm{x}} \textit{ by leg with constant speed of } v_{av_{leg\#x}} \\
\downarrow \\
\textit{conflicting points lies in a leg ?} \xrightarrow{N} \textit{no feasible LO} \\
\downarrow Y \\
\textit{determine heading from level turn algorithm} \\
\downarrow \\
LO = ds \sin \Delta\psi \\
\downarrow \\
\textit{lateral offset} = LO
\end{gathered}$$

3.4.2 Vertical constraint derivation

In the vertical domain, an altitude constraint (ALT) was identified in the literature (Section 2.4.3). According to the description of the altitude constraint derived from the Smiths FMCS guide, an altitude constraint could be applied to waypoints in the trajectory. The altitude constraint consists of a flight level following by a suffix ”A”, ”B” or no suffix, meaning ”at or above”, ”at or below” or ”at” respectively. Two altitude values indicate an altitude ”window” or block crossing restriction for that waypoint. Applying an altitude constraint is a way of notifying the aircraft the solution space in the vertical domain in which it can generate its trajectory.

The altitude constraint is a relative easy to be determined constraint in comparison with the other spatial constraints. The strategy which is used for the altitude constraint is to allocate the aircraft another altitude at the conflicting time instances. The altitude constraint is applied to the first and continues to the last conflicting point in the trajectory. This is achieved by considering the first and last conflicting point as auxiliary waypoints on which the altitude constraint is applied on. If a waypoint lies between the conflicting points, this waypoint is automatically assigned the same altitude constraint as applied to the first and last conflicting point. Application of the altitude constraint is illustrated in Figure 3.7. The aircraft is allowed to acquire any other altitude 1000 ft higher or lower than the flight level at the conflicting time instances.

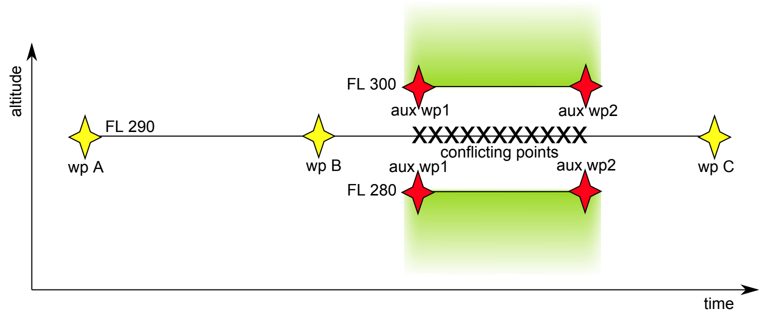

Figure 3.7: Side view of a trajectory where the concept of using the Altitude constraint is illustrated. The aircraft is allowed to acquire any Flight Level at or above 300 or at or below 280 for that part of the trajectory where the conflict is predicted.

In pseudo code, the strategy for determining the altitude constraint is:

altitude constraint:
determine the first and last conflicting point = aux A, aux B
↓
for all conflicting points: find the lowest altitude = L
for all conflicting points: find the highest altitude = H
↓
constraining auxiliary wp:
aux A, aux B: "At or above" flightlevel(L + 1000 ft)
or
"at or Below" flightlevel(L - 1000 ft)

3.4.3 Time shift constraint derivation

Besides spatial constraints, time shift constraints were also identified in the literature study (Section 2.4). As mentioned earlier, a time shift constraint restricts the aircraft in the time domain and thus affects the timing profile of the 4D trajectory. Time shift constraints can be used to resolve dynamic conflicts. Applying a time shift constraint to a static conflict, e.g. a conflict between an aircraft 4D trajectory and Special Use Airspace, does not resolve the conflict. The speed constraint (SP) and required time of arrival constraint (RTA) were identified as possible constraints (Section 2.4.2) and are discussed in more detail below.

Speed constraint

According to the description of the speed entry as stated in the Smiths FMCS guide [21], the speed entry is applied to a leg and restricts the aircraft to maintain a specified speed during the whole leg. The speed entry in a leg can be seen as a way to delay[1] the aircraft in order to resolve the conflict.

According to this definition of the speed entry, no speed range can be given as input to the FMS. The speed entry restricts the aircraft to maintain a certain speed within a feasible speed range. However, according to the vision of trajectory de-confliction in this work it is required to specify ranges in order to approximate the solution space. Through the use of constraints the solution space is notified to the aircraft instead of a single solution within the solution space. Therefore, the speed entry is not applied in this case study. The Required Time of Arrival provides the ability to specifies ranges, which is discussed next.

Required Time of Arrival constraint

According to the description of the RTA constraint derived from the Smiths FMCS guide, the RTA constraint is applied to a waypoint. As a consequence of applying a RTA constraint, the aircraft is restricted to be at the waypoint at a specified time. Besides the possibility of restricting the aircraft to be ”at” a certain time at the waypoint, the aircraft can also be restricted to be ”at or before” or ”at or after” a specified time at the waypoint.

A RTA constraint does not have the drawback of the speed constraints as described above. Where and when the delay (as a consequence of applying an RTA constraint) is absorbed is delegated to the aircraft FMS which construct the timing profile of the 4D trajectory taking the RTA constraint into account.

Given the description of the RTA constraint above, a strategy is needed which provides the appropriate time value for the RTA constraint. The analytical method discussed by Koeners [5] is used to derive the RTA constraint. This method provides the speed range which results in loss of separation and is outlined in more detail in Appendix A.

Given two aircraft predicted to be in conflict, the conflict can be resolved by delaying (slowing down) or advancing (speeding up) the maneuver aircraft (on which the constraint is applied) in order to pass behind or in front of the other aircraft respectively. The corresponding speed (range) that would result in the required minimal separation can have three outcomes:

1. no solution is found; meaning that an RTA constraint can not be computed, e.g. in a head-on conflict where the heading difference between both aircraft is 180 degrees,

2. only one speed is found which result in exactly the required separation; meaning an ”At” RTA can be computed,

[1]A delay can be positive or negative which corresponds to a slower or higher speed than nominal respectively.

3. there are two speed values at which the CPA is exactly the minimum required separation. All speeds between those values will result in CPA with a separation below the required minimum separation. For the lower speed value, an "At/After" RTA constraint is provided and for the upper speed value an "At/Before" RTA constraint is provided.

Suppose that the tool described by Koeners provides two speed values: a slow-down speed and a speed-up speed. If the aircraft would fly with a speed slower/equal than the slow-down speed or faster/equal than the speed-up speed, the CPA would be greater or equal than the minimal required separation. Given the path distance S of the trajectory up to the conflict point, the time t at the conflict point is given by:

$$t = S/V \tag{3.6}$$

where V is be the slow-down or speed-up speed (depending if an "At/After" RTA or "At/Before" RTA constraint is applied respectively). The difference between t and the original planned time t^* at the conflict point corresponds to the time delay which would result if the aircraft would fly the speed provided by the tool of Koeners. For a slow-down solution, an "At/After" RTA constraint can be applied and for a speed-up solution an "At/Before" RTA constraint with the corresponding time t at the conflict point can be applied.

The method to compute the speed range that result in a loss of separation assumes no turns before the conflicting points. If a trajectory has multiple turns before the conflicting points, the path length of the trajectory before the conflicting point can be treated as a straight path in order to obtain the speed range that would result in loss of separation. Furthermore, a requirement of this approach is that both original trajectories have constant speed and track. The approach can therefore be applied in cases when the conflicting points lies inside a leg of both the trajectories. The conflicting legs can then be treated as a trajectory (i.e. discarding the other parts of the trajectories) on which the approach described by Koeners is applied on. Thus in order to determine the RTA, the method described by Koeners is used in the case the conflicting points lies within a leg for both trajectories involved and when the speed in the corresponding legs are constant.

In pseudo code, this strategy is written as:

RTA constraint (strategy 1: constant track and speed):

conflicting points lies in a leg ? $\xrightarrow{N}$ *use strategy 2*

$\downarrow$ Y

conflicting legs have constant speed and track ? $\xrightarrow{N}$ *use strategy 2*

$\downarrow$ Y

Koeners approach:

derive slow-down speed $= V_{sd}$ *and speed-up speed* $= V_{su}$

calculate the time difference between t^* *and* t *at the first conflicting point*

if time difference $>$ *0 :RTA "At/Before" @ first conflicting point*

if time difference $<$ *0 :RTA "At/After" @ first conflicting point* *if Koeners approach provides 1 speed value: RTA ”At” @ first conflicting point*

When the conflicting points lies in two (or more) consecutive legs with different track and speed the approach is not feasible and another strategy has to be followed. In this case the time delay to determine the RTA constraint is estimated using the approach described in Appendix B which is similar to the one discussed by Idris et al. [37]. Appendix B provides a method to determine the time delay Δt required to resolve a predicted conflict. This is done by transforming the conflict into a time-space *(t-s)* domain. The time space-domain provides a framework to visualize and analyze the conflict between two aircraft in time and space and as a consequence to determine the time shift required to resolve the conflict.

Given the estimated time delay obtained from the approach as described in Appendix B, a trial trajectory is constructed and tested for conflict to ensure that the estimated time delay resolves the conflict. This is required since this approach of determining the time delay from the space-time approach is not as accurate as the analytical approach by Koeners. If the estimated time delay is not feasible, the estimated time delay is stepwise increased for a positive time delay (corresponding to ”At/After” RTA) or stepwise decreased for a negative time delay (corresponding to ”At/Before” RTA).

In pseudo code, this second strategy applicable to all cases is written as:

RTA constraint (strategy 2):
get estimated time delay from time-space model
↓
estimation check:
stepwise increase estimated time delay with unit time step: ”At/After” RTA
stepwise increase estimated time delay with unit time step: ”At/Before” RTA

3.5 Implementation

In the Section 3.2 of this Chapter the Concept of Operation was discussed. In the Concept of Operation a flowchart is presented which is used in the design process of the system which generates constraints for conflicting 4D trajectories. In Section 3.3 and Section 3.4 the conflict detection and constraint generation function are discussed respectively. This Section discusses the application of the approach as discussed in the preceding sections of this Chapter. As stated in the Introduction of this Chapter, the goal of the case study was to explore if the derived constraints and corresponding methods are feasible for conflict resolution of 4D trajectories. Therefore conflict scenarios were used on which the derived constraints and approaches are applied on to see whether the approaches result in constraints values (ranges which are used to approximate the solution space).

The design and derivation of the constraints are presented in the previous Sections using the flowchart as guideline. The system accepts as input the requested (user preferred) 4D trajectories and resolves detected conflicts by generating constraints in the spatial and time shift domain. The function blocks, as illustrated in the flowchart, are implemented in MATLAB R2013a.

3.5.1 Conflict scenarios

As mentioned earlier in this Chapter, conflict scenarios were found on the NASA website [22] which contains large numbers of conflict scenarios. In this work, the scope is restricted to demonstrate the concept of trajectory de-confliction through the use of constraints and simulation of large numbers of conflict scenarios is of less importance. Ten conflict scenarios obtained from the NASA website were chosen on which the approach is applied on. To keep this document readable and to avoid repetition, a randomly chosen conflict scenarios is discussed in this document.

The conflict scenarios include the 4D trajectories of two conflicting aircraft. Arbitrary conflict scenarios were chosen and resolved in the manner discussed in this chapter. The conflicting trajectories obtained from the NASA website contains the aircraft location in time which is captured by its latitude, longitude (in degrees) and altitude (in feet above the earth geoid[2]) coordinates. The latitude and longitude coordinates are transformed to the Cartesian coordinate system with the crossing point of the Greenwich meridian and the equator as origin, X-axis pointing North and Y-axis pointing 90 degrees to the right of the X-axis.

For the conflict scenarios with two aircraft involved, the constraints will typically be specified by a range with a lower bound and an "infinite" upper bound. In practice, the situation may occur that resolution of a scenario with two aircraft could result in a conflict with another aircraft trajectory which was not initially involved in the conflict. A "non-infinite" upper bound is then required to specify the feasible range and therefore guaranteeing that the resolution of the initial conflict does not incur a secondary conflict with another aircraft trajectory. In order to determine if an upper bound can be specified in the approach outlined in this Chapter, an intruder aircraft is added to the conflict scenarios obtained from the NASA website. The intruder aircraft initial is not in conflict with the other aircraft trajectories, but is used to see if an upper bound can be determined.

In the approach discussed in this Chapter the following principals are used:

- Radius and height of the protected zone are known (5 nm and 1000 ft respectively); if another aircraft is within the protected zone of the other aircraft, the minimum separation criteria are violated and a conflict is detected.
- ATM is delegated with the task of conflict detection and resolution while the aircraft is responsible for trajectory generation. Data-link is required and available for air-ground communication; air-ground communication is needed for down-linking of the trajectories and up-linking of the constraints.
- 3D positions and time at the positions of the aircraft involved are known and available in the form of a 4D trajectory.
- Resolution occurs prior to execution (offline); the requested trajectory is generated prior to execution, conflict detection and resolution are also performed prior to execution.

[2]Geoid: the particular equipotential surface that coincides with mean sea level and that may be imagined to extend through the continents.

- Resolution is performed by imposing constraints. Constraints are specified by a range (lower bound and upper bound) and are used to approximate the solution space.
- Resolution, i.e. imposing constraints, is applied on one of the aircraft involved in the conflict which is also known as the maneuver aircraft.
- Aircraft follow their planned trajectory accurately; the on-board FMS is capable of flying the 4D trajectory accurately, weather models are assumed to be perfect.
- Weather conflicts and environmental constraints (SUA) are omitted; only conflicts due to other aircraft are resolved.

In this thesis work the approaches are applied to conflict scenarios obtained from the NASA website of which one will be discussed in more detail in the next Chapter 4 to illustrate the approach and vision of using constraints in order to resolve conflicting 4D trajectories.

Results

4

In the previous Chapter the identified constraints were designed such that they are applicable according to the vision discussed in the Concept of Operation. The constraints are used to resolve conflicting trajectories by providing an approximation of the actual solution space. Methods to determine the appropriate constraints, i.e. constraints values defined by a lower and upper bound, were derived. To validate this approach, an first analysis is made in a case study on which the approach is applied on conflict scenarios to determine whether constraints values could be found. This chapter presents the results of applying the approach, discussed in the previous Chapter, by illustrating it using a conflict scenario.
This chapter is outlined as follows. In Section 4.1 the resolution of the conflict scenario and the corresponding characteristics are discussed. Section 4.2 ends this chapter with a summary.

4.1 Resolutions

The system as described by the flowchart in Section 3.2 accepts as input 4D trajectories and provides upon detection of a conflict constraints as output. The constraints and approaches used to acquire the constraint values are outlined in the Section 3.4. The approaches are implemented using MATLAB R2013a and applied on conflict scenarios obtained from the NASA website [22]. One arbitrary conflict scenario is discussed in this document which illustrates the approach of conflict resolution through the use of constraints. The conflict scenario initially involves two aircraft trajectories in conflict. An intruder aircraft trajectory is added to the scenario in order to determine if ”non-infinite” upper bounds could be specified as described in Section 3.5.1.

Relevant figures related to the resolution of the conflict scenario are presented. The aircraft trajectories were obtained from the NASA website. The aircraft on which the constraints are imposed on is chosen arbitrary and corresponds to the trajectory colored in red.

For this conflict scenario no lateral offset (LO) constraint is provided because the conflicting points are spread over multiple legs (see Section 3.4.1 for the strategy used to determine a lateral offset constraint). For the remaining type of constraints the ATO, ALT and RTA are found and will be discussed next. The lateral offset constraint is illustrated using another conflict scenario where a feasible lateral offset constraint is found and is discussed in Section 4.1.5.

4.1.1 Representation of the conflict scenario

FIGURE 4.1:
The conflict involves conflicting trajectories of aircraft ASH2680_1 (represented by the blue line) and aircraft COM442_F_1 (represented by the red line).

FIGURE 4.2:
The same conflict scenario projected on the ground and is displayed in Figure 4.2. The conflict occurs at common Flight Level 290 (8839 m). The conflicting points of the maneuver aircraft COM442_F_1 are displayed with an asterisk in the Figure.

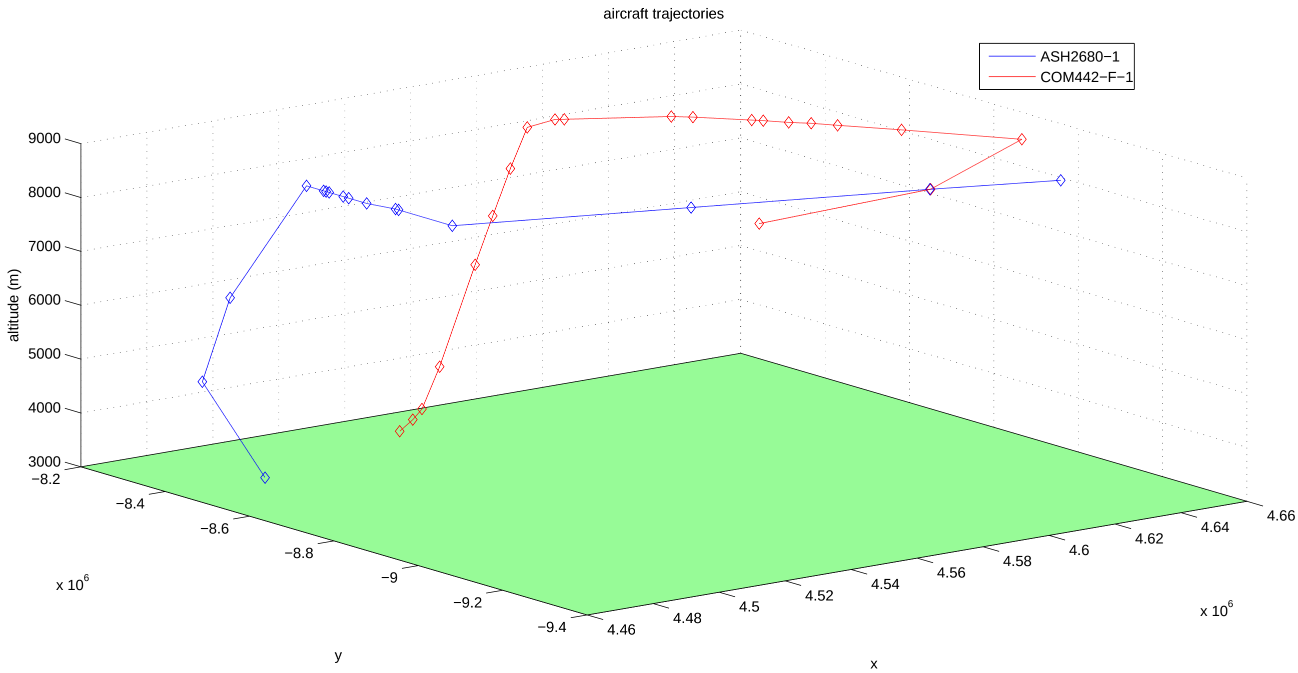

Figure 4.1: The conflict scenario with trajectories of aircraft ASH2680_1 and aircraft COM442_F_1

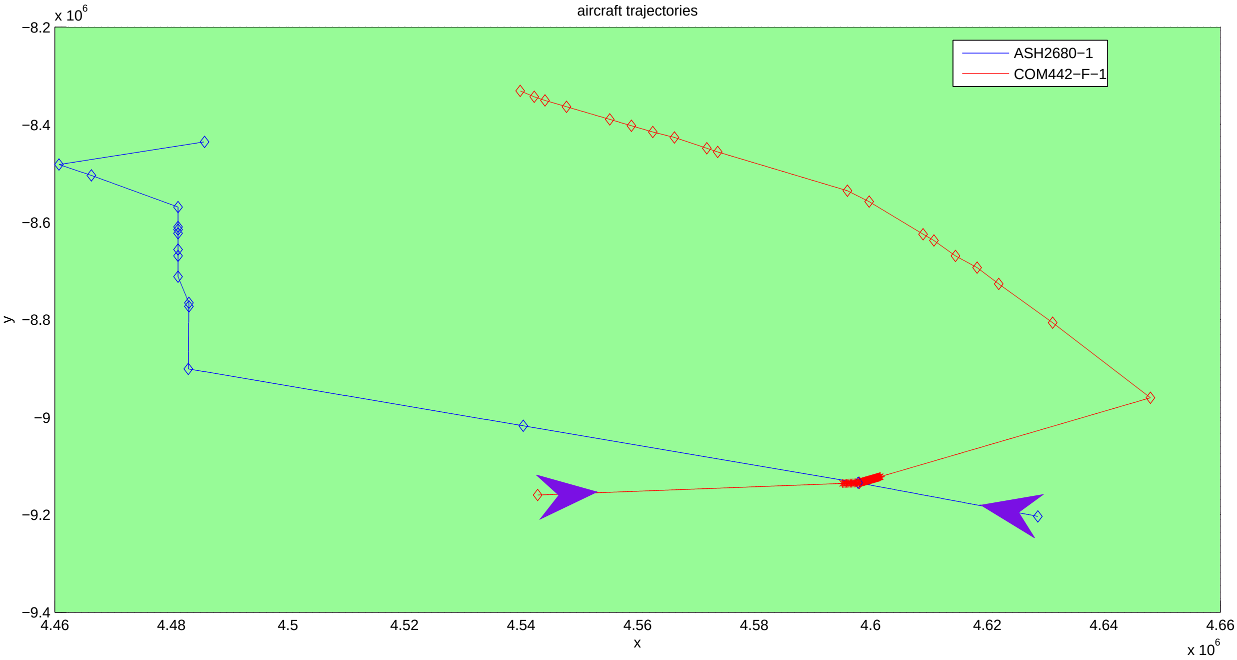

Figure 4.2: Top view of the aircraft trajectories projected on the ground involved in the conflict scenario. The conflicting points of the maneuver aircraft trajectory are marked with asterisks.

4.1.2 Resolution in the lateral domain

FIGURE 4.3:
In Figure 4.3, the minimum negative along track offset is illustrated. This minimum negative offset, where the second waypoint displayed in the Figure is shifted backwards along the trajectory, is required to resolve the conflict with the blue aircraft trajectory. According to the design strategy of the along track offset constraint, the maximum allowable offset for a negative ATO can be as long as the leg length of the leg prior to the waypoint on which the offset is applied. The feasible range, in which the waypoint can be shifted negatively along track, is thus specified by the minimum offset (lower bound) and the maximum offset which corresponds to the first waypoint of the red trajectory displayed in the Figure.

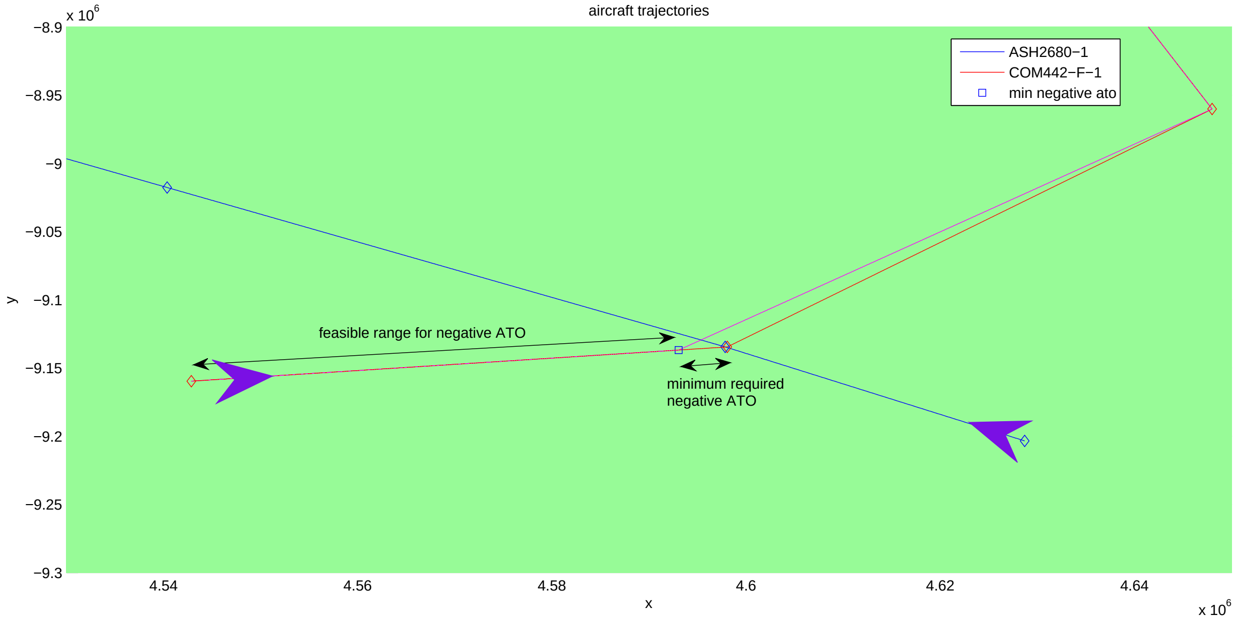

Figure 4.3: The minimum offset required to shift the second waypoint backwards along trajectory is indicated together with the feasible range of along track offsets which results in resolving the conflict with the blue trajectory.

FIGURE 4.4:
As discussed in the Section 3.5.1, an intruder aircraft is added to the conflict scenarios obtained from the NASA website. This intruder aircraft is added to see if an upper bound can be defined which is not "infinite"[1]. In Figure 4.4 the intruder aircraft trajectory is represented by the black trajectory. The intruder aircraft is initially not in conflict with either the red or blue aircraft trajectory. As a consequence of the trajectory of the intruder aircraft the feasible range for the negative along track offset is changed, as indicated in the Figure. Any negative offset within the feasible range resolves the primary conflict, i.e. the conflict between the red en blue trajectory, and does not incurs a secondary conflict with the intruder aircraft. If an along track offset larger than the upper bound would be applied, a conflict with the intruder aircraft will be introduced and an along track offset smaller than the lower bound would not resolve the primary conflict between the blue and red trajectory. A solution (in the lateral domain) is then to restrict the maneuver aircraft (i.e. the aircraft corresponding to the red trajectory) using a negative along track offset constraint which is specified by the feasible range as indicated in Figure 4.4.

[1] In case of the ATO, the upper bound is finite and defined by the maximum offset which, according to the FMCS guide, can be as long as the leg length of the leg prior to the waypoint on which the offset is applied (for a negative offset constraint) or as long as the leg length of the leg after the waypoint on which the offset is applied (for a positive offset constraint (See also Section 3.4.1).

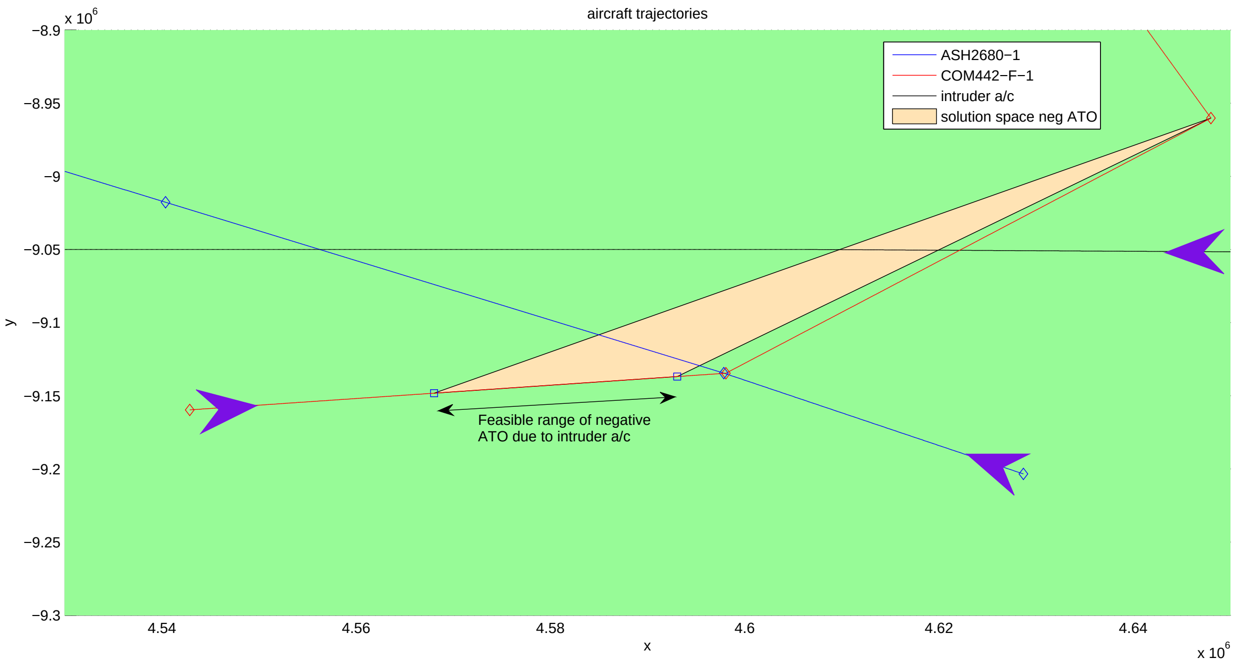

Figure 4.4: The feasible range of along track offset constraint is changed due to the introduction of an intruder aircraft which trajectory was initially not in conflict with the blue or red trajectory.

FIGURE 4.5:
In Figure 4.5 the same approach is followed as in the case of the negative ATO, but in this case the second waypoint of the red trajectory is shifted forwards along track. The minimum positive along track offset required to resolve the conflict is illustrated in the Figure and similar to the negative ATO, the maximum allowable positive offset can be as long as the leg length of the leg after the waypoint on which the offset is applied. This range of feasible positive along track offset is illustrated in the Figure.

Figure 4.6:
As in the case of the negative ATO, the feasible range of the *positive* along track offset constraint is changed due to the introduction of an intruder aircraft. The trajectory of the intruder aircraft, indicated by the black line in the Figure, was initially also not in conflict with the blue or red trajectory. Any positive offset within the feasible range resolves the primary conflict, i.e. the conflict between the red en blue trajectory, and does not incurs a secondary conflict with the intruder aircraft. If an along track offset larger than the upper bound would be applied, a conflict with the intruder aircraft will be introduced and a along track offset smaller than the lower bound would not resolve the primary conflict between the blue and red trajectory. Another solution (in the lateral domain) is then to restrict the maneuver aircraft (i.e. the aircraft corresponding to the red trajectory) using a *positive* along track offset constraint which is specified by the feasible range as indicated in Figure 4.4.

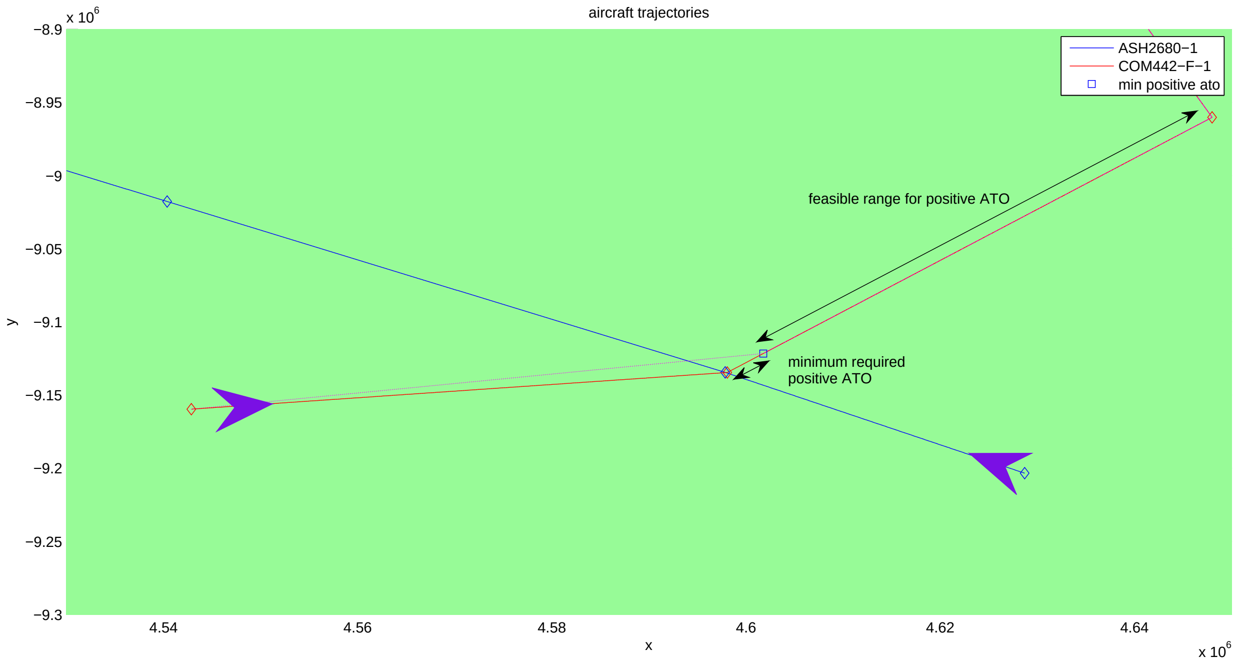

Figure 4.5: The minimum offset required to shift the second waypoint *forwards* along trajectory is indicated together with the feasible range of along track offsets which results in resolving the conflict with the blue trajectory.

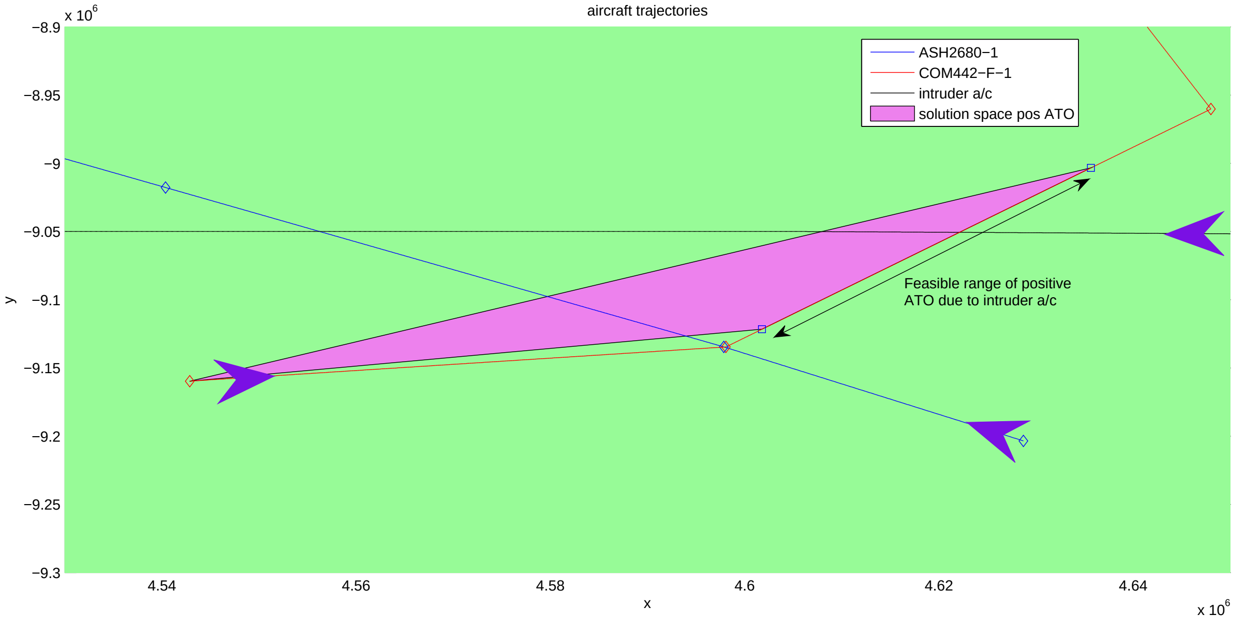

Figure 4.6: As in the case of the negative ATO, the feasible range of the *positive* along track offset constraint is changed due to the introduction of an intruder aircraft. The trajectory of the intruder aircraft was initially not in conflict with the blue or red trajectory.

4.1.3 Resolution in the vertical domain

Figure 4.7 and Figure 4.8

In Figure 4.7 a side view of the same aircraft trajectories depicted in Figure 4.1 are illustrated. At the vertical axis the altitude in Flight Level is given and at the horizontal axis the time in seconds. Between time instance 273 and 293 the minimum required separation is violated which is indicated by the conflicting points marked with asterisks in the Figure. From the Figure it is also clear that the conflict occurs at a common altitude, i.e. Flight Level 290. Figure 4.8 illustrates the same trajectories but zooms into the relevant area where the conflict occurs.

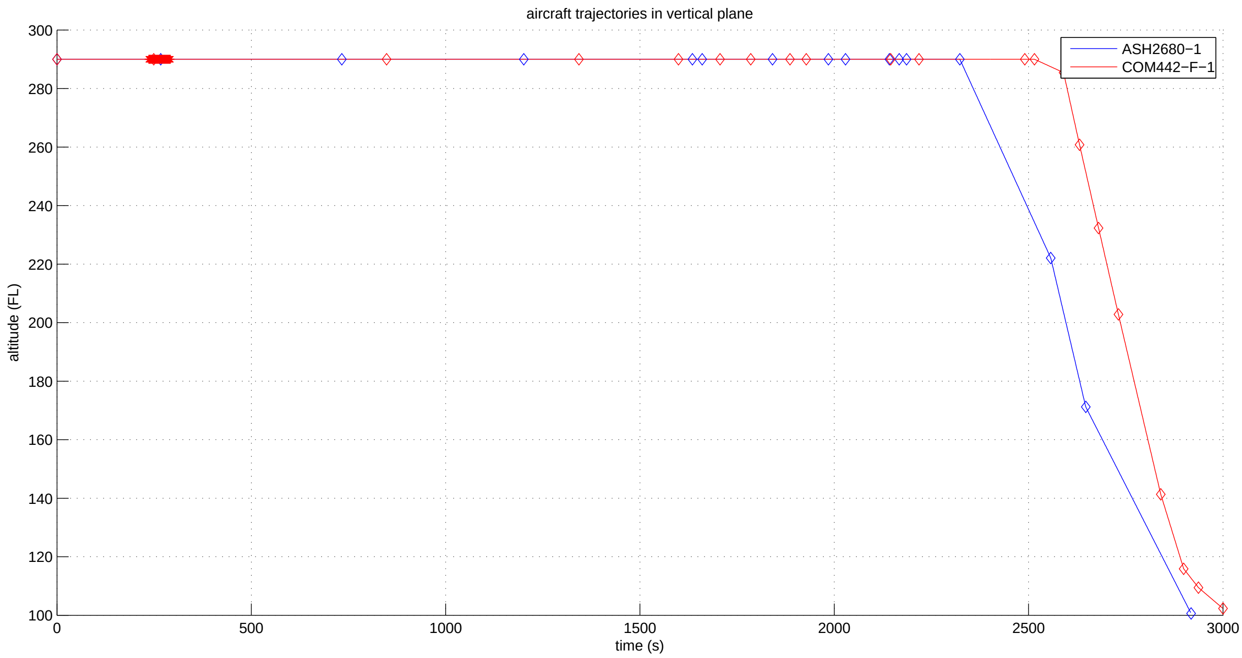

Figure 4.7: Vertical view of the aircraft trajectories. The conflicting part of the trajectories are indicated by the asterisks.

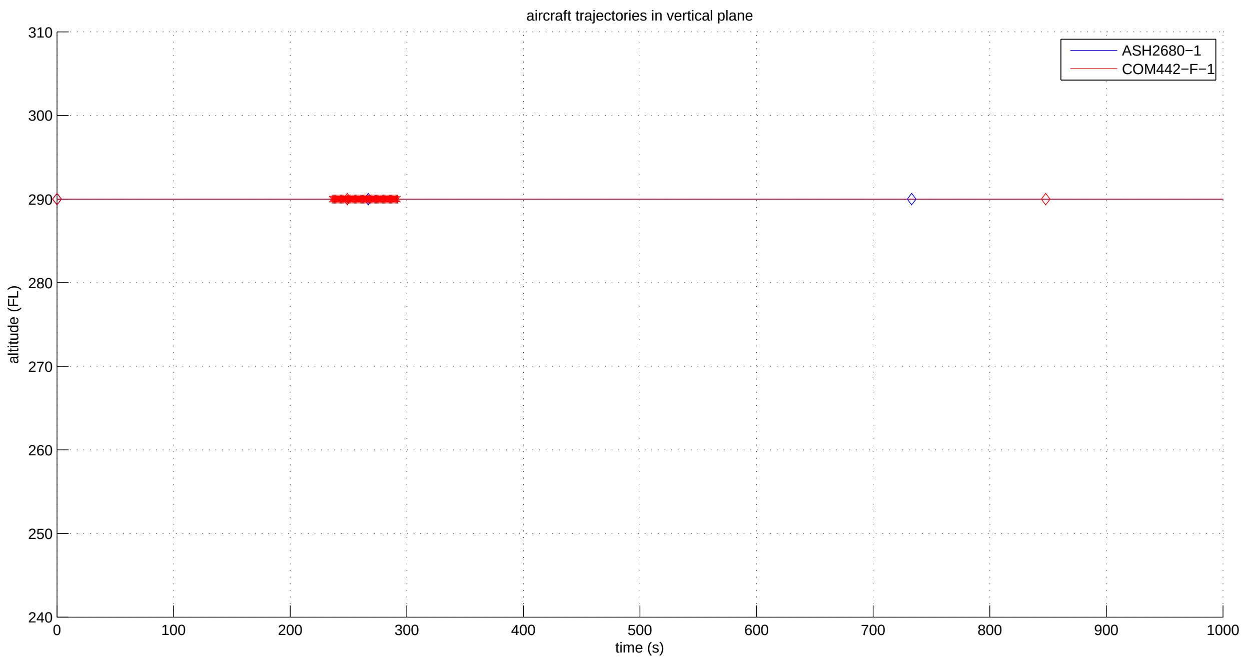

Figure 4.8: Vertical view of the aircraft trajectories, zoomed into the conflicting part.

FIGURE 4.9
Resolution in the vertical plane is performed by applying an Altitude constraint. The altitude constraint consists of a flight level following by a suffix ”A”, ”B” or no suffix, meaning ”at or above”, ”at or below” or ”at” respectively. Two altitude values indicate an altitude ”window” or block crossing restriction. In this case, the maneuver aircraft is not allowed to be at Flight Level 290 for the time instances that the conflict is predicted. This is indicated in the Figure by an block above and under the conflicting part of the trajectory which has a height of 1000 ft. A solution in the vertical domain is thus to apply an altitude constraint, in the form of a block crossing restriction, which is specified by FL 300 and FL 280, applied at the first conflicting point in the trajectory and is maintained until the last conflicting point. Any altitude outside this block may be acquired in the regeneration of the revised trajectory by the maneuver aircraft.

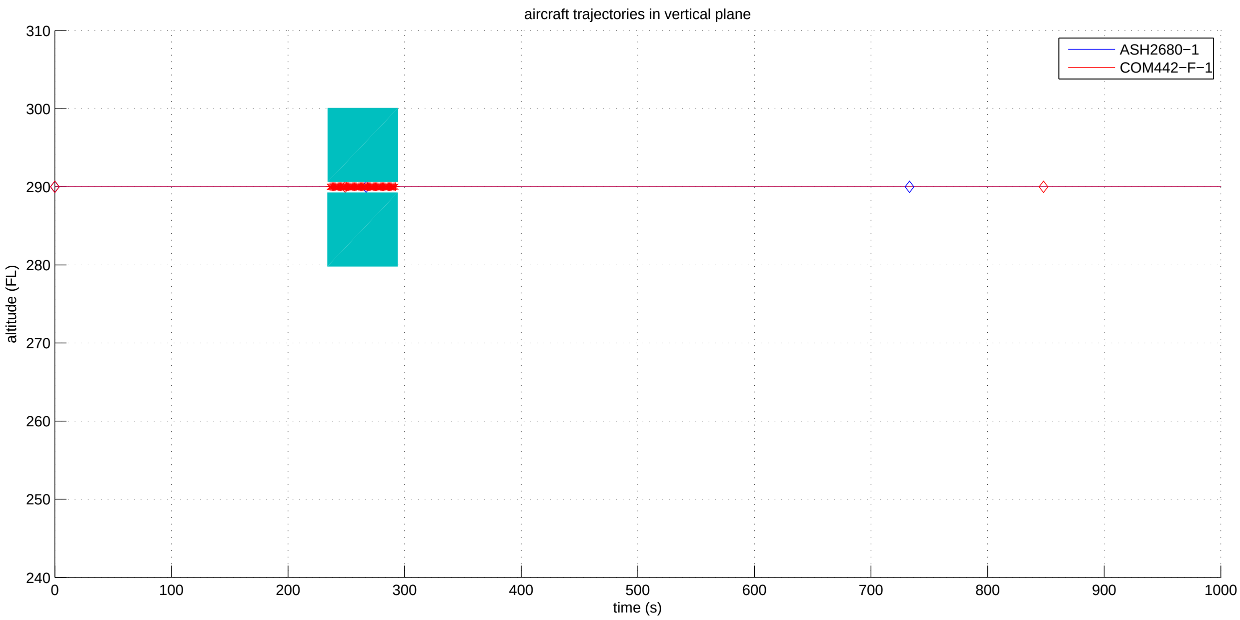

Figure 4.9: Altitude where the maneuver aircraft is not allowed to operate during the time instance when the conflict is predicted. This is indicated by the altitude block in the Figure which it is not allowed to enter.

FIGURE 4.10
In practice, the maneuver aircraft may comply to the imposed block restriction constraint as discussed above. However, it may acquire an altitude which would result in incurring a secondary conflict. This scenario is evaluated by also introducing an intruder aircraft to the conflict scenario. The intruder aircraft trajectory is illustrated in Figure 4.10 by the black trajectory. In order to determine if additional constraining of the maneuver aircraft is required (in the vertical domain) in order to avoid a secondary conflict with the intruder aircraft, the maneuver aircraft trajectory is situated at the same Flight Level for all time instances as that of the intruder aircraft and conflict detection is performed. This resulted in that the maneuver aircraft is not allowed to operate at the same Flight Level (or within 1000 ft above or below) between time instances 338 and 391 (i.e. the maneuver aircraft would then within 5 nautical mile of the intruder aircraft horizontally and thus also the horizontal separation requirement would be violated). In the Figure, this is indicated by the conflicting points along the intruder aircraft trajectory and block crossing restriction for that part. The additional constraint, due to the presence of the intruder aircraft, is an additional block crossing restriction defined by Flight Level 270 and 250 starting at that point along the maneuver aircraft trajectory at time 338 and is maintained until the point along the maneuver aircraft trajectory at time 391. The first block crossing restriction resolves the conflict with the blue trajectory whereas the second block restriction prevents a secondary conflict with the intruder aircraft. Any trajectory which lies outside the two blocks may generated by the maneuver aircraft.

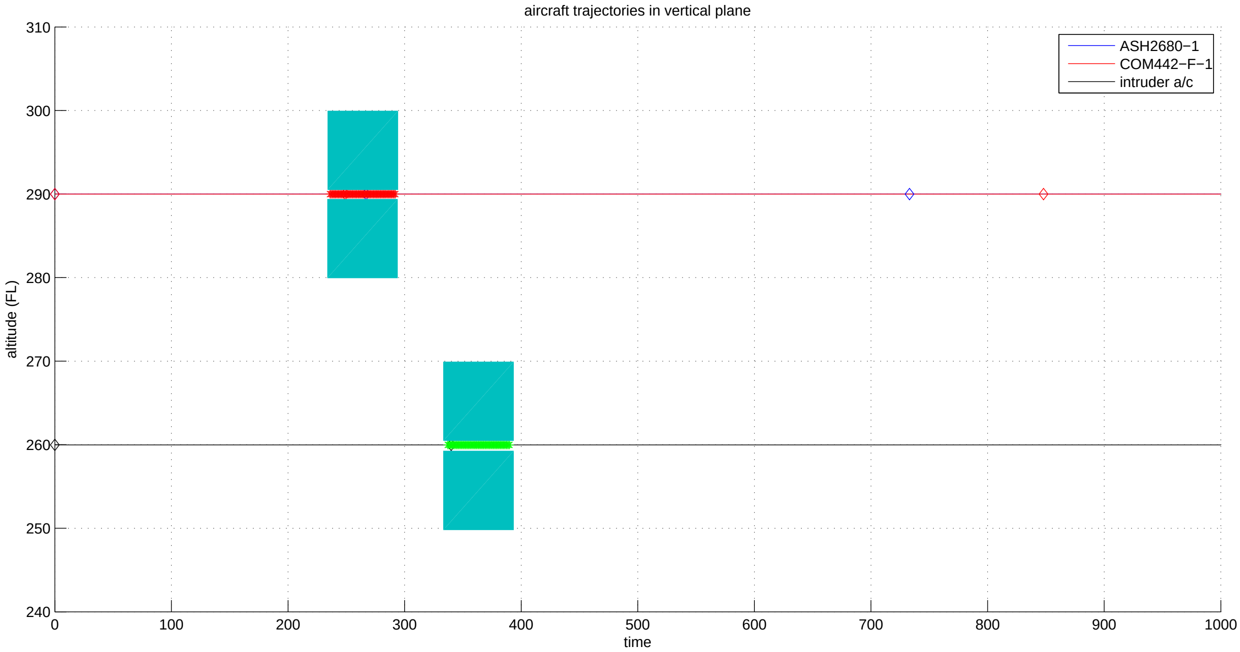

Figure 4.10: Due to the presence of the intruder aircraft, the maneuver aircraft is constrained by imposing two block crossing restrictions on it. This is indicated in the Figure by the colored blocks where its trajectory is not allowed to fall within those blocks.

4.1.4 Resolution in the time domain

FIGURE 4.11 AND FIGURE 4.12
If both aircraft involved in the conflict would travel along its trajectory a conflict is predicted to occur. This is illustrated in Figure 4.11 where the minimum horizontal separation requirement is violated between time instance 237 and 293. Since the vertical separation requirement is also violated for this part of the trajectory, as illustrated in Figure 4.1, a conflict is thus detected. Resolution in the time domain consists of applying an RTA constraint as has been discussed in Section 3.4.3. The conflict between the aircraft represented by the blue and red trajectory is resolved by applying an RTA constraint ”AT/BEFORE 17 seconds” or an ”AT/AFTER 80 seconds” constraint at the first conflicting point. This implies that the maneuver aircraft, represented by the red trajectory, is required to arrive *at least* 17 seconds earlier or 80 seconds later than initially planned at that point along the trajectory which corresponds to the first conflicting point. This is also indicated in Figure 4.12 where the horizontal distance of the ”AT” restriction is illustrated which results in just complying to the minimal horizontal separation requirement. The solution space in the time domain consists thus of any time profile of the 4D trajectory which complies to the imposed RTA constraint. That means that the aircraft has to speed up (for the AT/BEFORE constraint) or slow down (for the AT/AFTER constraint) in the first leg of its trajectory in order to meet the RTA constraint.

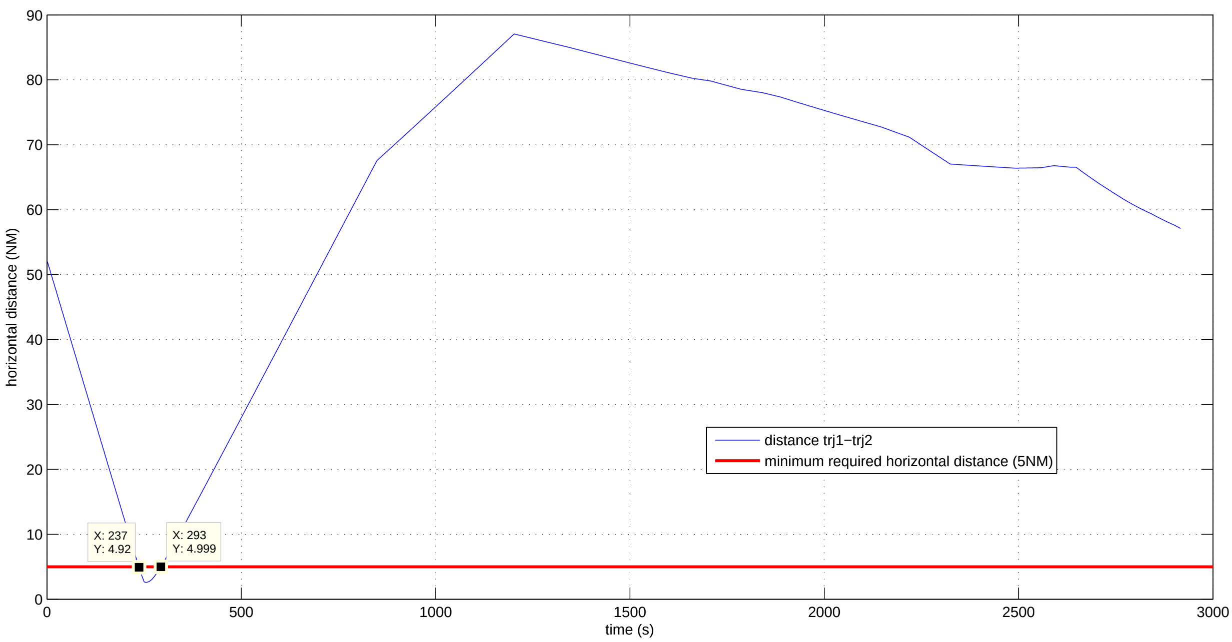

Figure 4.11: Illustration of the violation of the minimum horizontal separation requirement of the aircraft trajectories.

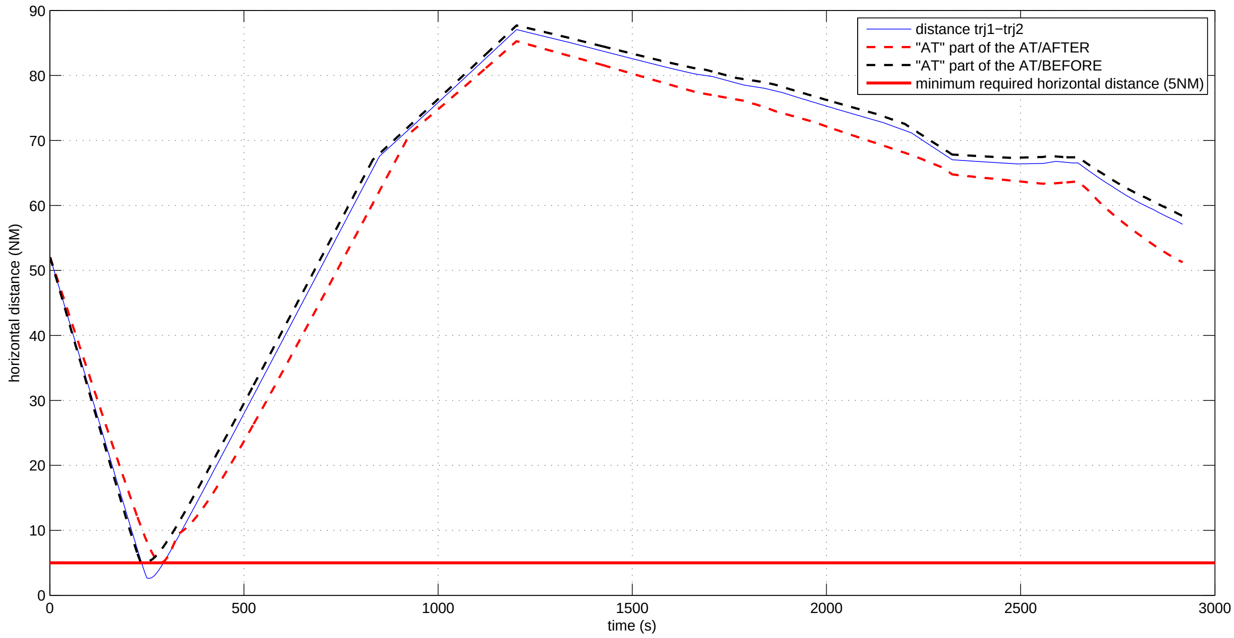

Figure 4.12: Illustration of the minimum required time delay needed to resolve the conflict. This minimum required time delay corresponds to the "AT" part of the RTA constraint and would result in the aircraft just misses the protected zone of each other.

Figure 4.13 and Figure 4.14 and Figure 4.15
If no other aircraft trajectory would be involved, defining a lower bound is sufficient in order to resolve the conflict. The feasible range is then defined by the lower bound corresponding to the ”AT” part of the constraint and ”infinite” upper bound corresponding to the ”BEFORE” or ”AFTER” part of the constraint. In practice this additional constraining may be needed in order to avoid secondary conflicts with other aircraft. This is illustrated by adding an intruder aircraft to the scenario as illustrated in Figure 4.13. The intruder aircraft is initially not in conflict with either the aircraft corresponding to the red or blue trajectory as illustrated in Figure 4.14. As can be seen in the Figure 4.13, the intruder aircraft will follow part of the trajectory of aircraft COM442-F-1. The RTA ”AT/AFTER” constraint obtained to resolve the initial constraint between the red and blue trajectories could result in that the maneuver aircraft generates a time profile of its 4D trajectory which is in conflict with the intruder aircraft trajectory. The ”AT/AFTER” constraint requires that the maneuver aircraft, i.e. the aircraft corresponding to the red trajectory, obtains a lower speed profile in order to meet the RTA ”AT/AFTER” constraint. This ”slowing down” may be limited in order to avoid that the maneuver aircraft becomes in conflict with the in trail intruder aircraft behind. As a consequence, for these trajectories, additional constraining is required. The upper bound in this case is ”non-infinite” and is 240 seconds for the ”AT/AFTER”, i.e. the maneuver aircraft is required to be later than 80 seconds at the position in the trajectory corresponding to the first conflicting point, but not more than 240 seconds later than originally planned. Arriving 240 later at the point in the trajectory corresponding to the first conflict point, would just meet the separation requirement which is illustrated in Figure 4.15. For the ”AT/AFTER” case, no additional constraining is required since the intruder aircraft is in trail behind and speeding up (by at least the amount specified earlier to resolve the primary conflict, i.e. arriving 17 seconds earlier) would not result in a secondary conflict with the intruder aircraft.

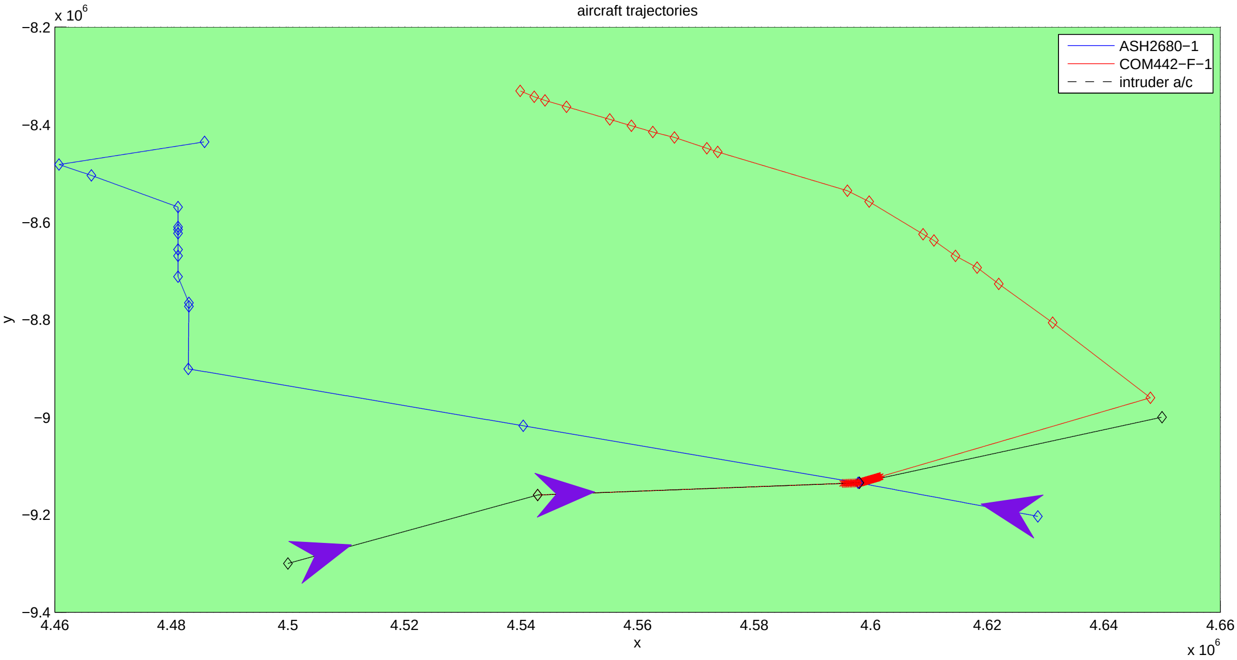

Figure 4.13: Adding an intruder aircraft to the scenario to illustrate additional constraining in the time domain.

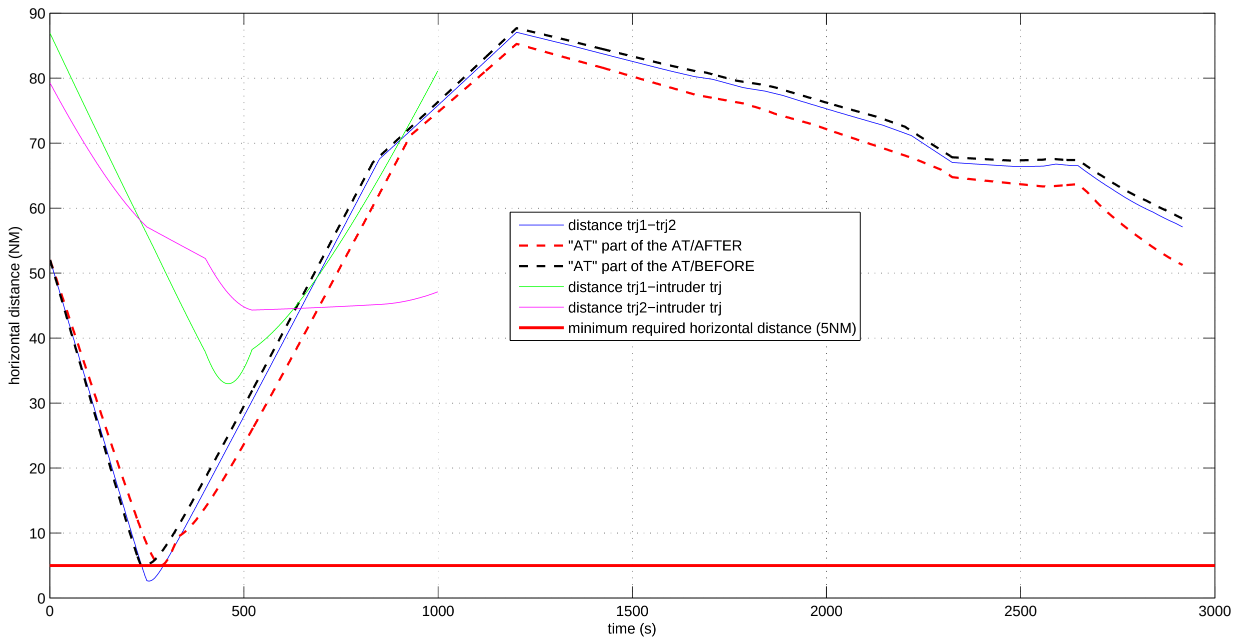

Figure 4.14: The intruder aircraft is initially not in conflict with the other aircraft trajectories.

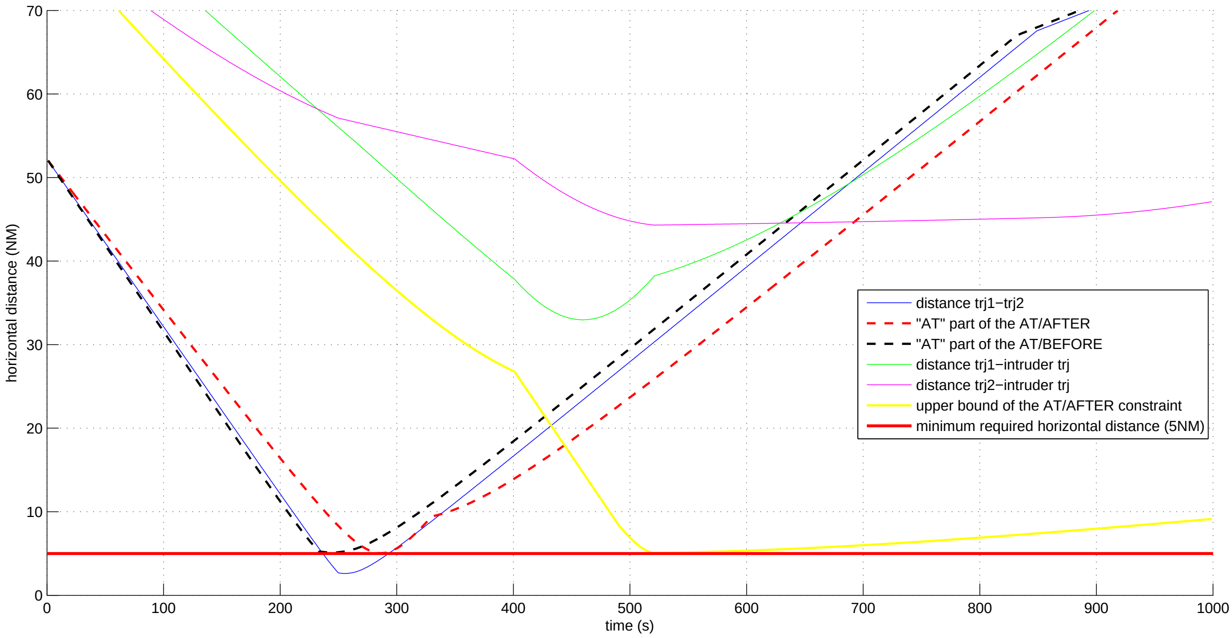

Figure 4.15: The upper bound for the RTA "AT/AFTER" constraint would result in just conforming to the minimum required separation.

4.1.5 Resolution using Lateral Offset

In the previous scenario no lateral offset constraint has been found. In order to illustrate the lateral offset constraint and for completeness the lateral offset constraint is illustrated in this Section using another conflict scenario where a feasible lateral offset constraint is found. To keep this document readable and omit repetition, the other feasible constraint for the scenario discussed in this Section are omitted.

Figure 4.16
Figure 4.16 illustrates a top view two aircraft in conflict and the corresponding conflicting part of the maneuver aircraft trajectory (red trajectory). Also illustrated in this Figure is the intruder aircraft trajectory. The intruder aircraft trajectory is not in conflict with either the red or blue aircraft trajectory. As mentioned earlier, the intruder aircraft is added to the conflict scenario to illustrated the solution space of the constraint.

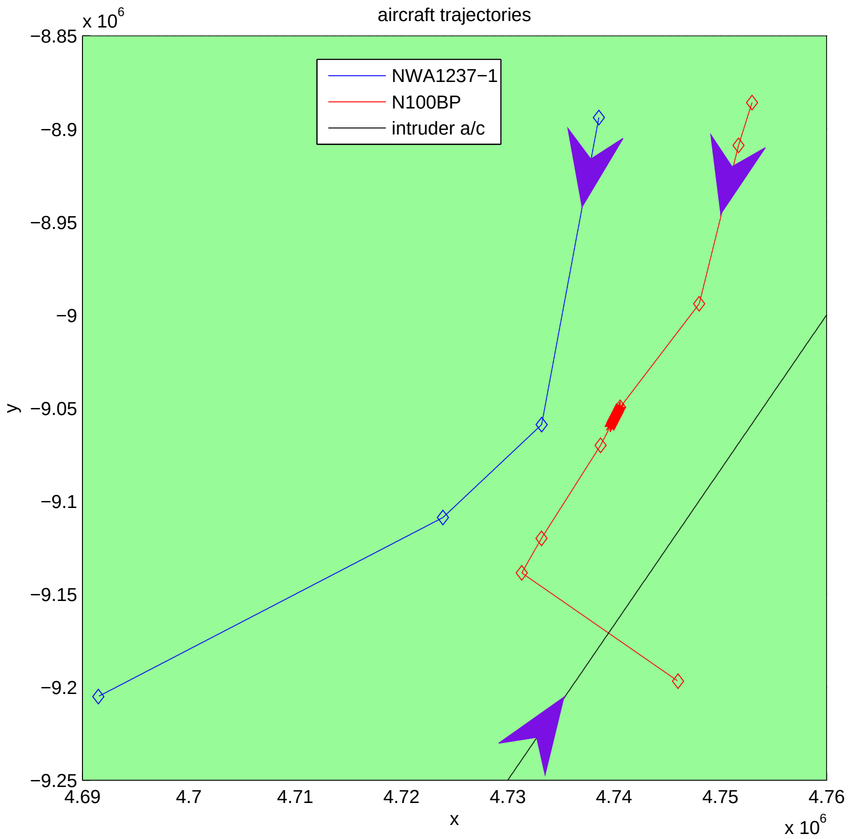

Figure 4.16: Top view of another conflict scenario used to illustrate conflict resolution through the use of the lateral offset constraint.

Figure 4.17

In order to resolve the conflict between the aircraft corresponding with the red and blue trajectories (which are initially in conflict), a minimum lateral offset of the fourth leg to the left is needed of 500 m. This is indicated in Figure 4.17 which zooms in the conflicting part of the trajectory. A lateral offset to the right of the maneuver aircraft trajectory was not found. The solution space for the left lateral offset is therefore defined by the range corresponding to the minimum left lateral offset (500 m) and an ”infinite” upper bound. However, in practice additional constraining may be required in order to avoid a secondary conflict with another aircraft. This is illustrated by adding an intruder aircraft to the scenario as mentioned before and is also illustrated in Figure 4.17. As a consequence of the presence of the intruder aircraft additional constraining is required, i.e. the solution space is defined by the range corresponding to an ”non infinite” upper bound. In this case the upper bound is 3500 m, i.e. the maximum lateral offset of the fourth leg of the maneuver aircraft trajectory is 3500 meter. The solution space for the left lateral offset is thus the range defined by the the minimum and maximum left lateral offset (500-3500 m). This is also illustrated in Figure 4.17.

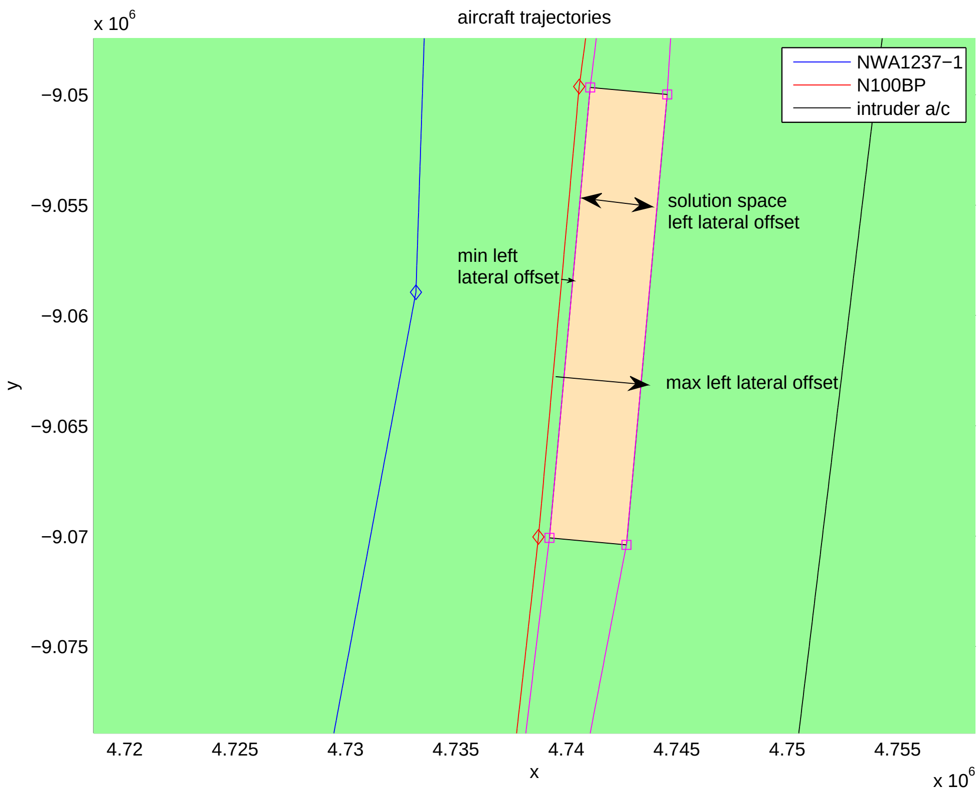

Figure 4.17: Figure 4.16 zoomed in the conflicting part of the trajectories. The solution space of the lateral offset constraint is illustrated.

4.2 Summary

Given the conflict scenarios obtained from the NASA ACC conflict resolution database [22], constraints are derived according to the strategies described in Chapter 3. In this Chapter the approach has been illustrated using a conflict scenario which involved two aircraft in conflict. The aircraft (trajectory) on which the constraint applied on was chosen arbitrary. For this particular scenario, constraints were found in the lateral, vertical and time domain and are captured in Table 4.1.

For this particularly conflict scenario no lateral offset constraint has been derived and only the Along Track Offset constraint was found. In the design strategy of the lateral offset constraint, it was considered that only a leg where the conflicting points lies within could be constrained with a lateral offset constraint. In the conflict scenario the conflicting points were spread over multiple consecutive legs and according to the design strategy, as described in Section 3.4.1, the lateral offset constraint could not be determined. Also described in Section 3.4.1 is a method of how a lateral offset could be applied to multiple consecutive legs. If this method would be feasible and applicable, it is expected that a feasible lateral offset constraints could be found.

In order to illustrate the lateral offset constraint and for completeness the lateral offset constraint is illustrated in Section 4.1.5 using another conflict scenario where a feasible lateral offset constraint is found.

Constraint	Description
ATO	negative ATO applied @ WP2 range: 3NM - 17.5 NM
ATO	positive ATO applied @ WP2 range: 7.3NM - 73.7 NM
LO	n/a
ALT	BLOCK CROSS: FL300 - FL280 for time 237 - 293
	and BLOCK CROSS FL270 - FL250 for time 338 - 391
RTA	AT/BEFORE -17 @ first conflict point
RTA	AT/AFTER +80 *and* AT/BEFORE +240 @ first conflict point

Table 4.1: Constraints

In the vision of using constraints for trajectory de-confliction, the constraints are used to approximate the solution space. As elaborated on in Section 3.2 it is required that the constraints defines a range. Within this range the aircraft is allowed to regenerate its trajectory upon detection of a conflict. Typically, the range is defined by a lower bound and upper bound. In the case that there are only two aircraft involved, the range is defined by an ”infinite” upper bound. However in practice an ”non-infinite” upper bound may be required as elaborated on in Section 3.5 due to the presence of other aircraft. In order to define this ”non-infinite” upper bound an intruder aircraft is added to the conflict scenarios obtained from the NASA database.

Determining the upper bound depends on the intruder aircraft trajectory, i.e. spatial geometry and time profile of the 4D trajectory. In the conflict scenario illustrated in this Chapter, the upper bounds were specified due to the presence of an intruder aircraft. In this thesis work the constraints, derived from the literature and the methods of how to determine the appropriate constraint values (the range) were applied to ten conflict scenarios. For these conflict scenarios at least one constraint was found which resolved the conflict even in the case the other aircraft involved was assigned to be the maneuver aircraft.

In the vision of this work the feasible constraints are provided to the aircraft by means of data-link communication. The aircraft determines from these feasible constraints the most appropriate one and corresponding revised trajectory (e.g. applies a constraint which results in the most economical trajectory or trajectory with the smallest time delay). Since the constraints provides an approximation of the solution space, of which the approximation lies within the actual total solution space, the constraint is considered to guarantee that the conflict is resolved while taken the assumptions into account.

5 Discussion and Future work

In the previous Chapter 4 a conflict scenario is discussed which is used to illustrate the approach and vision of conflict resolution through the use of constraints. The possible identified constraints found in the literature study were designed and strategies are derived to determine the appropriate constraint values. In the vision of using constraints in this work, the constraints provide an approximation of the solution space. This approximation is achieved by defining ranges for the identified constraints in the later, vertical and time domain and has been illustrated in the previous Chapter 4. In the design process and approach as described in Chapter 3 several assumptions were made. In reality, these assumptions has to be taken into account and cannot be omitted. This is further elaborated on in this Chapter followed by recommendations for future work.
This Chapter is outlined as follows. Section 5.1 discusses the assumptions made in this work and Section 5.2 provides recommendations for future work.

5.1 Discussion

The assumption is made that the minimum separation criteria is 1000 feet vertically and 5 nautical mile horizontally for all flight levels. Currently two vertical separation standards have been set. Because of the degradation of the pressure altimeter accuracy at high altitudes, the required vertical separation above FL290 is 2000 ft, whereas below FL290 the required vertical separation is 1000 ft [44]. In practice, the vertical separation of 2000 ft above FL290 can have influence on the constraint in the vertical domain, i.e the ALT constraint. The operational ceiling of allowable flight level will be reached sooner in case the required vertical separation is 2000 ft. As a consequence, the solution space of allowable flight levels will be decreased and thus the probability of finding a feasible ALT constraint will be lower than if the vertical separation was 1000 ft. This consequence may be enhanced by the fact that the optimal efficiency flight level for modern passenger aircraft is usually above FL290 [44] and users are thus eager to acquire higher flight levels. Consequently, the solution space for the ALT constraint is smaller and the risk of new conflicts is greater for trajectories operating above FL290.

Another assumption is that the 4D trajectories are available and that a trajectory includes the aircraft 3D position for every time step of 1 second. Conflicts with unequipped aircraft, i.e. aircraft which on-board FMS is not capable of providing a 4D trajectory, would be not detected. To cope with this, these aircraft could be assigned a certain airspace (certain flight levels where unequipped aircraft operates and) where the conflicts are handled in a manner as it is done nowadays (by the air traffic controller). Furthermore, larger step size could be acquired at a cost of higher uncertainty of the aircraft spatial position between the consecutive time steps. This could result in missed detections where aircraft are actually in conflict *between* two consecutive time steps but not *at* the time steps itself and as a consequence the requested trajectory passes through the conflict detection function. Unfortunately, there is no clear unambiguous agreement among the stakeholders about the 4D trajectory format as discussed in Section 3.3.1. According to the paper of Barnier et al. the largest allowable interval is a time step of 15 seconds [43]. In this work, the assumption is made that the aircraft (the on-board FMS) provides a 4D trajectory which is specified by waypoints and time at the waypoints. In order to cope with the phenomena of miss detections as described above, the assumption is made that the speed is constant between the waypoints and a 4D trajectory is constructed at the ATM side (in this work with a time step of 1 second as explained in Section 3.3.1).

Conflict detection and resolution are performed offline (prior to execution). In the ideal case the active trajectories are always free of conflict. However, conflict between active trajectories could arise when an aircraft in flight request an alternative 4D trajectory which deviates from the agreed 4D trajectory. These alternative 4D trajectories could be requested if for example the aircraft wants to avoid unexpected weather cells. If conflicts between active trajectories needs to be resolved it is required that the resolution initiation horizon is long enough. Furthermore coordination is needed like when the requested trajectory will become active and perhaps some penalty function in order to avoid arbitrary requests.

Weather conflicts are omitted but should be taken into account in practice. Weather conflict should be dealt with as late as possible but prior to execution of the trajectory. Weather forecasts of days or longer ahead may not be as reliable as actual weather forecasts.

In this work the conflict scenarios involved two aircraft where one of them was assigned to be the maneuver aircraft. In practice, an aircraft may request a trajectory which is in conflict with more that one aircraft trajectory. Some strategy has to be determined in order to resolve the conflict. A strategy could be to resolve the first conflict which occurs in time. As a consequence of the first constraint, which resolves the first conflict occurring in time, the other conflict may be resolved. If this is not the case additional constraining is required.

The trajectories involved in the conflict scenarios included instantaneous transition of track and speed between the legs. In practice, this should be smooth transitions. A function is then needed to transform abrupt changes into smooth transitions. Subsequently the strategies to determine the constraints can be applied to these trajectories, as has been done for the conflict scenarios of which one is illustrated in Chapter 4.

5.2 Future work

As mentioned earlier and discussed in the Concept of Operation, it is required that the constraint specifies ranges. As discussed in Section 3.4.1 the Along Track Offset and Lateral Offset does not provide a range (according to their definition derived from the Smiths FMCS guide [21]). However, preliminary results from the case study showed that the these constraints may become feasible by small adjustments to existing FMS, i.e. the availability of specifying ranges for the ATO and LO, and thus could be used for conflict resolution in the vision of this work. It is noteworthy to investigate other types of constraints which, by small adjustments like in the case of the ATO and LO, may be applicable according to the approach described in this work.

Furthermore, the assumption was made that one of the aircraft involved in the conflict was assigned to be the maneuver aircraft, i.e. the maneuver aircraft is the aircraft on which the constraints are imposed on. In scenarios where the solution space is ”small” or if even no solution can be found to resolve the conflict cooperative resolutions can remedy the situation. Cooperative resolution may increase the possibility in finding a solution to the conflict at the expense of increased complexity of the approach used to determine the solution.

For the conflict scenarios on which the derived constraints and methods to determine the constraints are applied on, always a constraint was found which resolved the conflict. Suppose that no solution is found for the conflicting trajectories even in the case cooperative resolution is applied as described above. A deferral strategy could then be used where the resolution function waits a specified amount of time and then tries from scratch. The rationale behind this is that maneuvering space may become available as other aircraft modifies or even cancel their trajectories. Of course, this strategy does not guarantees that a solution is found. Another option would be to reject the requested trajectory (without imposing constraints) and thus forces the aircraft to generate a new 4D trajectory after which the resolution process is started again.

In this work, single solution constraints are used to illustrate approach and vision of trajectory de-confliction through the use of constraints. However, a combination of different types of constraints may also resolve the conflict like for example applying both an RTA and Along Track Offset constraint.

This work mainly focused on illustrating conflict resolution through the use of constraints according to the vision discussed in the Concept of Operation in Chapter 3. Simulation involved small numbers of conflict scenarios. The derived constraints and methods were applicable to the conflict scenarios, i.e. in all cases at least one feasible constraint was found. However, the approach is still far from complete, as can be noted from the above, and a comprehensive algorithm is needed which is generic and applicable to all conflict scenarios taking the above discussion points and future work recommendations into account.

Summary and conclusion 6

In the previous Chapters of this thesis document, the vision of using constraints for conflict resolution of 4D trajectories has been discussed. This Chapter provides a summary and concludes this thesis work.

6.1 Summary & Conclusion

In this thesis work the milestones defined in the Section 1.3 were used as guideline. Recapturing the motivation for this work. In the current ATC world user preferred routes are not prime considerations. Conflicts are predicted using known aircraft position and velocity vector. This approach has generally some uncertainty in the estimate of the future position of the aircraft and is referred as a state-based approach. Leaving the system of conflict detection and resolution as it is, namely air traffic controllers monitor aircraft and resolve predicted conflicts using vectors, seems not feasible since the system is already running at maximum and the air traffic is expected to grow 5% each year [7]. Several visions on future ATM operations exist to accommodate this increase of air traffic, e.g. NextGen and SESAR. A common function between the visions is 4D trajectory management. Rather than applying the state-based approach of conflict detection and resolution, trajectory information can be used to detect and resolve 4D trajectories a priori. This trajectory-based (sometimes referred as plan-based) operation is envisioned to play a key role in future Air Traffic Management.

Resolving conflicting 4D trajectories is one of the concerns in trajectory-based operations. A way is to use constraints to resolve conflicting 4D trajectories. The initial step in this thesis assignment was to perform a literature study in the field of existing de-confliction algorithms which resolve conflicts using constraints. The result of the literature showed that the focus in 4D trajectory management mainly relies on conflict resolution where the solution consists of trajectories (see Chapter 2). There are several remarks regarding this approach as has been discussed in Section 1.2.2. Alternatively constraints could be used for conflict resolution. In the vision of this work, applying constraints is a means of providing the aircraft the solution space in which it is allowed to regenerate a trajectory by itself rather than taking an imposed trajectory generated externally into account. This rationale is discussed in Section 1.2.2 where the aircraft is delegated with the task of trajectory generation and centralized ground based stations (Air Traffic Management) are responsible for conflict detection and resolution.

The philosophy of using constraints has already been raised by researchers like Wichman et al. [4]. However, the concept of using constraints for conflict resolution of 4D trajectories is insufficient dealt with in the literature. There is a lack of algorithms or approaches which provide constraints that resolve conflicting trajectories, i.e. which constraints to be used and how to determine the appropriate constraint value. The focus in the literature study therefore shifted to the search of constraints which could be applied in the vision of this work, i.e. resolving conflicts by providing an approximation of the solution space through the use of constraints. Possible constraints have been identified by looking at data-link capabilities and at state-of-the-art FMS input capabilities. This resulted in constraints in the lateral (Along Track Offset and Lateral Offset constraint), vertical (Altitude constraint) and time domain (Required Time of Arrival constraint). Corresponding methods which could be used to determine the appropriate constraint values were also derived.

In the vision of using constraints for trajectory de-confliction, the constraints are used to approximate the solution space. As elaborated on in Section 3.2 it is then required that the constraint defines a range. Within this range the aircraft is allowed to regenerate its trajectory upon detection of a conflict. Typically, the range is defined by a lower bound and upper bound. In the case that there are only two aircraft involved, the upper bound is defined by an ”infinite” range. However in practice an ”non-infinite” upper bound may be required as elaborated on in Section 3.5. In order to define this ”non-infinite” upper bound an intruder aircraft is added to the conflict scenarios.

In this thesis work the constraints, derived from the literature and the methods of how to determine the appropriate constraint values (the range) were applied in a first case study to ten conflict scenarios. The approach is implemented in MATLAB R2013a. For the conflict scenarios at least one constraint was found which resolved the conflict even in the case the other aircraft involved was assigned to be the maneuver aircraft. Since the constraints provide an approximation of the solution space of which the approximation lies within the actual total solution space, the constraint is considered to guarantee that the conflict is resolved while taken the assumptions into account. The approach of using constraints is illustrated in this work by presenting the resolutions of a arbitrary conflict scenario in Chapter 4.

This thesis work provides a contribution in the field of 4D trajectory management using constraints. Conflict resolution through the use of constraints has been explored in a case study according to the philosophy stated in the Concept of Operation. In particular, constraints applicable in this approach are identified and corresponding strategies are determined to obtain the appropriate constraint values. Given the results form the case study, preliminary conclusion can be made that it seems feasible to apply the identified constraints (by small adjustments, i.e. a range is required to specify the constraint by an lower and upper bound) and methods for conflict resolution of 4D trajectories, since the constraints are used to provide the aircraft an approximation of the actual solution space (where the approximation is required to be within the actual solution space). However, the approach discussed in this work is far from complete, as has been discussed in the previous Chapter 5 and a comprehensive algorithm is needed which is generic and applicable at all time.

Bibliography

[1] "U.s. centennial of flight commission." http://www.centennialofflight.gov/essay/Government_Role/Air_traffic_control/POL15.htm.

[2] austro control. http://www.austrocontrol.at/en/content/atm/lotse/taetigkeit/taetigkeit.shtml.

[3] S. Bradford. http://www.icao.int/inexses/Presentation/Day1/1c_3_Bradford_Merkle.pdf.

[4] K. Wichman, L.G.V, Lindberg, L. Kilchert, and O. Bleeker, "Four-dimensional trajectory based air traffic management," 2004.

[5] J. Koeners, "Exploring benefits of a separation assessment tool to achieve level 3 sa for speed advisories."

[6] Eurocontrol, "Fans-1/a operations manual," 2006.

[7] Eurocontrol. http://www.eurocontrol.int/corporate/public/faq/about_us_faq.html#qa12.

[8] Eurocontrol, "Eurocontrol long-term forecast: Flight movements 2010-2030." https://www.eurocontrol.int/sites/default/files/content/documents/official-documents/forecasts/long-term-forecast-2010-2030.pdf.

[9] C. of Operations for the Next Generation Air Transportation System. http://www.faa.gov/nextgen/; www.jpdo.gov/library/NextGen_v2.0.pdf.

[10] SESAR. http://www.eurocontrol.int/sesar/public/subsite_homepage/homepage.html.

[11] T. Prevot, "Nextgen technologies for mid-term and far-term air traffic control operations."

[12] R. A. Paielli, "Trajectory specification for high-capacity air traffic control," 2005.

[13] J. Krozel, M. Peters, K. Bilimoria, C. Lee, and J. Mitchell, "System performance characteristics of centralized and decentralized air traffic separation strategies," 2001.

[14] H. Erzberger, "The automated airspace concept," 2001.

[15] F. official forecast of aviation activity at FAA facilities. http://aspm.faa.gov/main/taf.asp.

[16] M. Standar. http://www.icao.int/inexses/presentation/day1/1c_1_Standar.pdf.

[17] J. Krozel and M. Peters, "Strategic conflict detection and resolution for free flight," 1997.

[18] K. Wichman, G. Carlsson, and L. Lindberg, "Flight trials: 'runway-to-runway' required time of arrival evaluations for time-based atm environment," 2001.

[19] A. F. M. S. German Aerospace Center (DLR). http://www.dlr.de/fl/en/desktopdefault.aspx/tabid-1127/1591_read-3004/.

[20] H. Idris, R. Vivona, and T. Al-Wakil, "Trajectory-orientated approach to managing traffic complexity," 2009.

[21] Smiths, "Fmcs guide."

[22] NASA. http://airtrafficconflictresolutions.arc.nasa.gov/aac/index.html.

[23] C. L. Tallec and A. Joulia, "Ifats: 4d contracts in 4d airspace," *Infotech@Aerospace 2007 Conference and Exhibit*, pp. AIAA 2007–2778, 2007.

[24] D. t. d. Eurocontrol ATM Lexicon. https://extranet.eurocontrol.int/ and http://atmlexicon.eurocontrol.int/en/index.php/4D_Trajectory.

[25] J. K. Kuchar and L. C. Yang, "A review of conflict detection and resolution modeling methods," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, no. 4, pp. 179–189, 2000.

[26] R. Irvine, "The gears conflict resolution algorithm," 1998.

[27] R. Vivona, D. Karr, and D. Roscoe, "Pattern-based genetic algorithm for airborne conflict resolution," 2006.

[28] T. Farley and H. Erzberger, "Fast-time simulation evaluation of a conflict resolution algorithm under high air traffic demand," 2007.

[29] H. Erzberger, "Automated conflict resolution for air traffic control," 2006.

[30] G. Granger, N. Durand, and J.-M. Alliot, "Optimal resolution of en route coniňĆicts," 2001.

[31] H. Idris, D. Wing, R. Vivona, and J. Garcia-Chico, "A distributed trajectory-orientated approach to managing traffic complexity," 2007.

[32] C. Tomlin, G. Pappas, and S. Sastry, "Conflict resolution for air traffic management: A study in multiagent hybrid systems," 1998.

[33] M. S. Deby and W. E. K. III, "Free flight separation assurance using distributed algorithms."

[34] G. Dowek and C. Muñoz, "Conflict detection and resolution for 1,2,...n aircraft," 2007.

[35] G. Dowek, C. Muñoz, and V. Carreño, "Provably safe coordinated strategy for distributed conflict resolution," 2005.

[36] E. Frazzoli, Z. Mao, J. Oh, and E. Feron, "Resolution of conÂŔflicts involving many aircraft via semidefinite programming," 2001.

[37] H. Idris, T. Hsu, R. Vivona, and S. Green, "Time-based conflict resolution algorithm and application to descent conflicts," 2003.

[38] Eurocontrol, "Eurocontrol cpdlc." http://www.eurocontrol.int/agdl/public/ standard_page/BACKGROUND_INFORMATION.html.

[39] "Fmc user's manual."

[40] E. Ernst, "Flight management computer (fmc) manual."

[41] A. Kaiser, "Fms guide."

[42] R. Bach, C. Farrell, and H. Erzberger, "An algorithm for level-aircraft conflict resolution," 2007.

[43] N. Barnier and C. Allignol, "4d-trajectory deconfliction through departure time adjustment," 2009.

[44] D. Manolakis and C. Lefas, "Aircraft geometric height computation using secondary surveillance radar range differences," 1994.

Appendix A

Koeners [5] developed a speed evaluation tool which supports the controller in assessing future separation if current speeds remain constant. If two aircraft have a 3D conflict, the speed of both aircraft determines wether a loss of separation will occur. By computing the Closest Point of Approach (CPA) as a function of trial speeds the speed range can be determined analytically that resloves the conflict. This appendix presents this approach as discussed by Koeners [5].

Separation between two vehicles can be described as a function of the starting positions and the change vectors. A change vector can be described as a function of speed and time. Figure A.1 shows an example of two vehicles with a separation R.

The speed vectors of the vehicles in the x and y direction can be described as:

$$dx_1 = v_1 \cdot \sin(track_1) \tag{A.1}$$

$$dy_1 = v_1 \cdot \cos(track_1) \tag{A.2}$$

$$dx_2 = v_2 \cdot \sin(track_2) \tag{A.3}$$

$$dy_2 = v_2 \cdot \cos(track_2) \tag{A.4}$$

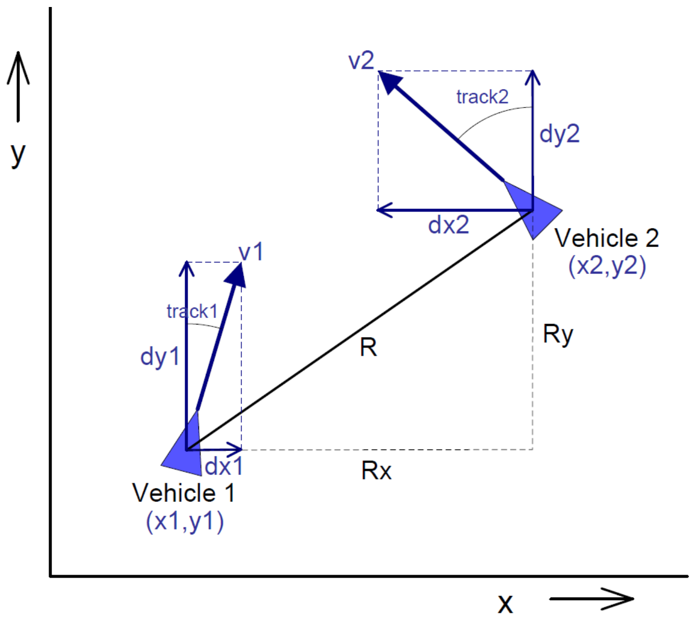

Figure A.1: Example separation between two vehicles [5]

Given the starting positions the separation between the vehicles in the x and y direction can be described as a function of time and speed:

$$Rx = (x_1 - x_2) + (dx_1 - dx_2) \cdot t \tag{A.5}$$

$$Ry = (y_1 - y_2) + (dy_1 - dy_2) \cdot t \tag{A.6}$$

The separation as a fuction of time and speed follows from:

$$R = \sqrt{Rx^2 + Ry^2} \tag{A.7}$$

The separation between two vehicles as a function of time for a certain speed can now displayed, see Figure A.2.

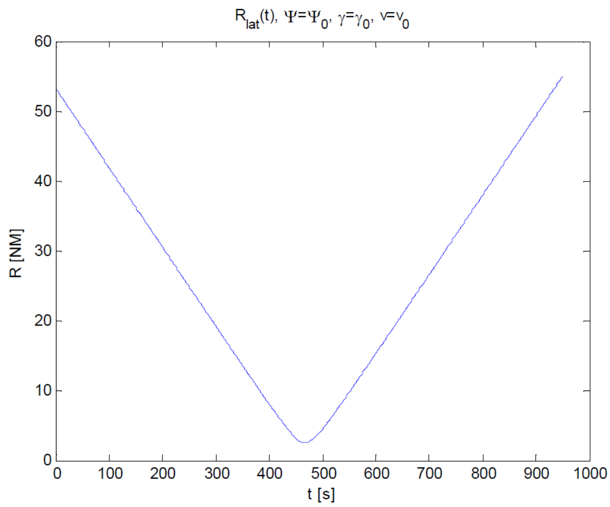

Figure A.2: Example separation as a function of time [5]

The minimum separation is below the minimum required separation. In this example the speed would result in a separation of 0.5 nmi at the CPA. To find a speed that would result in a CPA with the required separtion, a function is needed where speed is the only variable. If the required separation and the speed of one of the vehicles are known, the speed of the other vehicle that would result in the required minimal separation can be computed (Figure A.3).

There can be three different outcomes. The first outcome is that for a given speed of one of the vehicles no speed of the other vehicle can be found which result in a CPA with the required separation. The second outcome is that only one speed is found which results in a CPA with exactly the required separation. The third outcome gives two speeds at which the minimum separation is exactly the required separation (Figure A.3). All speeds in between will result in a CPA with a spearation below the required separation.

The approach outlined above assumes that the involved vehicles have a conflict geometry similar to Figure A.1. When there are more turns before the conflict, the path length can be computed up to the last turn before the conflict and treated as if it was a straight path.

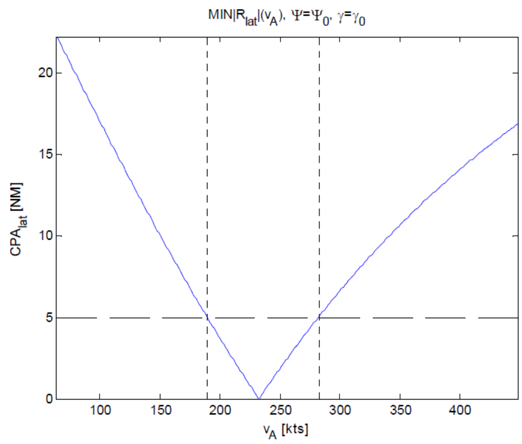

Figure A.3: Example separation at CPA as a function of speed [5]

Appendix B

This appendix provides the method similar to the one discussed by Idris et al. [37] which determines the time shift Δt required to resolve a predicted conflict.

The approach discussed by Idris et al. assumes no turns during the predicted conflict. This approach is used to make an initial guess of the time shift Δt which would resolves the conflict. The availability aircraft information in the 4D trajectory (positions and time) makes it possible to apply the same approach discussed by Idris et al. to conflict scenarios without using the formulas in Idris' paper. As a consequence the assumption of no turns during the predicted conflict can be omitted. Disadvantage of this approach is that the time shift Δt can not be calulated analytically but have to be estimated.

The implementation of the estimated time shift Δt does not guarantee that the conflict between the two aircraft is resolved. Stepwise increment of the time shift is a possibility to deal with this issue.

The approach of estimating the time shift Δt is as follow:

- Given two trajectories in conflict, the first step is to assign a maneuver aircraft and an intruder aircraft. The maneuver aircraft performs the maneuver (i.e absorbs the time shift Δt) while the intruder aircraft continues as planned. The maneuver aircraft first position is taken as reference point *(x_0,y_0)* (horizontal conflicts are assumed, therefore z-position is omitted).

- Subsequently the line s which measures the distance from the reference point is determined for the maneuver aircraft.

- Next, the intersection points of the circle with radius of 5 nautical mile and the position of the maneuver aircraft as center for every conflicting time instance is determined. This is done with the function:

  ```
  intersections(X1,Y1,X2,Y2)
  ```

  The function accepts as input two curves represented by their X- and Y-vector and gives as output the *x,y* intersection points of the curves.

- The distance between the reference point and the intersection points are calculated. The result is a locus of points *s1* and *s2* which can be visualized in the *(s-t)* plane.

- The time shift Δt can be estimated by shifting the line s upwards (positive time shift) or downwards (negative time shift).

This M-code for this approach is given below and is applied for the conflict scenarios. A time shift Δt=-17s or Δt=+80s results in a feasible time shift. The corresponding RTA can then be determined by At/Before -17 or At/After 80 constraint applied at the first conflicting point which is discussed in Chapter 4.

```
%Maneuver aircraft: trj2 COM442-F-1
%Intruder aircraft: trj1 ASH2680-1

%Minimum required horizontal separation: 5nm=9260m
%Reference point is begin point of trj2: (trj2.x(1),trj2.y(1))

%Conflicting pnts: cd1 are indices where trj1 is in conflict, same for trj2
[cd1,cd2]=conflict_detection_incl_t(trj1,trj2);

s=sqrt((trj2.x-trj2.x(1)).^2+(trj2.y-trj2.y(1)).^2);

%find the points of intersection of the circle around intruder a/c (trj1)
%with line s==trj2.
for i=1:length(cd1)
    X1=trj1.x(cd1(i))+R*cos(0:0.01:2*pi);
    Y1=trj1.y(cd1(i))+R*sin(0:0.01:2*pi);
    [X0,Y0]=intersections(X1,Y1,trj2.x,trj2.y);
    smatrix(i,:)=[X0(1) Y0(1) X0(2) Y0(2)];
end

s1=sqrt((smatrix(:,1)-trj2.x(1)).^2 +(smatrix(:,2)-trj2.y(1)).^2 );
s2=sqrt((smatrix(:,3)-trj2.x(1)).^2 +(smatrix(:,4)-trj2.y(1)).^2 );

% estimate the time delay using estimate_delay function
% input to the function is maneuver aircraft trj, line s,
% conflicting points of the trajectories and sign
%(1 for pos time delay, -1 for neg time delay)
pdt=estimate_delay(trj2,s,cd1,cd2,s1,s2,1);
ndt=estimate_delay(trj2,s,cd1,cd2,s1,s2,-1);

%------------function: estimate_delay-------------------------------------
% determine pos (sign=1) and neg (sign=-1) time delay using tangent lines.
function [dt]=estimate_delay(trj2,s,cd1,cd2,s1,s2,sign)
   if sign>0
        for i=1:length(s1)
            s1_hoek(i)=tan(trj2.t(cd1(i))/s1(i)); %bepalen van de raaklijn
        end

        ind1=find(s1_hoek==max(s1_hoek));
        tijd1=trj2.t(cd1(ind1));
```

```
        s1value=s1(ind1);
        verschil=abs(s-s1value);
        verschil_min=min(verschil);
        tijds1_ind=find(verschil_min==verschil);
        tijds1=(trj2.t(tijds1_ind));
        dt=tijd1-tijds1;

    elseif sign<0
        for i=1:length(s2)
            s2_hoek(i)=tan(trj2.t(cd2(i))/s2(i)); %bepalen van de raaklijn
        end

        ind2=find(s2_hoek==min(s2_hoek));
        tijd2=trj2.t(cd2(ind2));
        s2value=s2(ind2);
        verschil=abs(s-s2value);
        verschil_min=min(verschil);
        tijds2_ind=find(verschil_min==verschil);
        tijds2=(trj2.t(tijds2_ind));
        dt=tijd2-tijds2;

    end
end
```

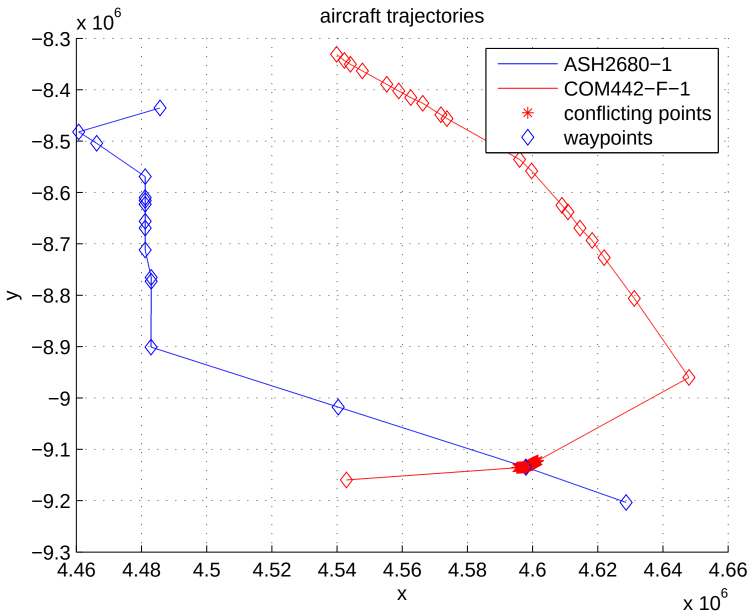

Figure B.1: Horizontal conflict of Conflict #1

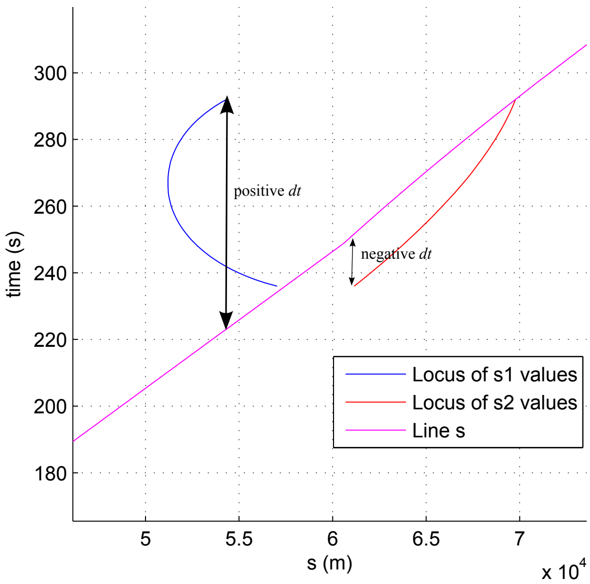

Figure B.2: Line *s* plotted in the conflict set defined by the locus of values of *s1* and *s2*

Appendix C

This appendix provides formulas for obtaining the heading for the maneuver aircraft which result in just miss of the protected zone of the intruder aircraft. The formulas are derived from the level turn algorithm discussed by Bach et al. [42].

Considering two aircraft in conflict with known trajectories and taking the maneuver aircraft relative to the other aircraft, the relative trajectory of the maneuver aircraft with respect to the other (stationary) aircraft is obtained. If the aircraft are in conflict, i.e. the minimum separation requirement is violated, the relative trajectory of the maneuver aircraft would penetrate the protected zone of the other aircraft. In Figure C.1 this relative geometry is given. It is clear that aircraft 1 would penetrate through the protected zone (represented by the circle with radius $R = 5$ nmi) and their will be loss of separation if both aircraft would continue along their trajectory as planned.

The relative velocity of aircraft 1 is given by

$$V_r = V_1 - V_2 \tag{C.1}$$

In a Cartesian coordinate system with x-axis pointing North and y-axis pointing East, the track of the relative velocity vector is then given by

$$\psi_R = \tan(N/E) \tag{C.2}$$

where

$$N = V_1 \sin\psi_1 - V_2 \sin\psi_2 \tag{C.3}$$

$$E = V_1 \cos\psi_1 - V_2 \cos\psi_2 \tag{C.4}$$

are the components of the velocity vector of the maneuver aircraft along the x and y axis respectively. In order to stay out of the PZ of aircraft 2, a change of track of the relative velocity vector V_r is needed. The new track of the relative velocity vector ψ_R^* resulting in a just miss of the PZ of aircraft 2 is given by

$$\psi^*_{R_{LH}} = \psi_R - (\beta - \alpha) \text{ for a left hand turn,} \tag{C.5}$$

$$\psi^*_{R_{RH}} = \psi_R + (\beta + \alpha) \text{ for a right hand turn.} \tag{C.6}$$

The angle α is the angle between the relative velocity vector V_r and the Line of Sight from aircraft 1 to aircraft 2, $Slos$, and is given by

$$\alpha = \psi_R - \psi_{Slos} \tag{C.7}$$

where

$$Slos = \sqrt{\Delta x^2 + \Delta y^2} \tag{C.8}$$

$$\psi_{Slos} = \tan^{-1}(\Delta y/\Delta x) \tag{C.9}$$

The angle β is the angle between the *Slos* and the line corresponding to the new relative velocity vector which extrapolation is tangent to the PZ of aircraft 2. The angle β is given by

$$\beta = \pm \sin^{-1}(R/Slos) \tag{C.10}$$

Now all parameters are know to derive the track of the new relative velocity ψ_R^* in equation (C.5). The velocity of the maneuver aircraft, aircraft 1 in the example, is kept unchanged since the size is kept constant and the velocity vector is rotated about point C. Applying the Law of Sines, the new track of aircraft 1 is determined by the following formula

$$\psi_1^* = \psi_R^* + \sin^{-1}[\sigma_v \sin(\psi_2 - \psi_R^*)]; \sigma_v = V_2/V_1 \tag{C.11}$$

This formula is valid when $V_2 \leq V_1$. If $V_2 > V_1$ then there are two solutions for both a left hand turn and a right hand turn. The point representing aircraft 1 in the relative geometry is then placed outside the circle with center C. Therefore the extrapolation of the relative velocity vector which is tangent to the PZ of aircraft 2 intersects the circle in two points. In this case the second solution, besides the one obtained from equation (C.11), is given by

$$\psi_1^* = \psi_R^* - \sin^{-1}[\sigma_v \sin(\psi_2 - \psi_R^*)] + 180^o \tag{C.12}$$

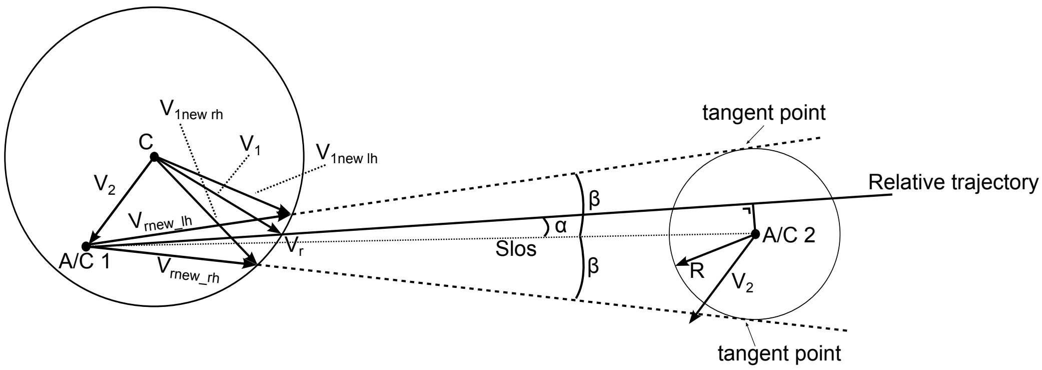

Figure C.1: Relative geometry of two aircraft in conflict