Co-operative Conflict Resolution in Autonomous Aircraft Operations Using a Multi-agent Approach

by Miguel Angel Vilaplana Ruiz

A thesis submitted to the Faculty of Engineering of the University of Glasgow for the degree of Doctor of Philosophy

All aspects of the work presented herein are original in concept except where indicated

Abstract

The future Air Traffic Management is anticipated to feature Autonomous Aircraft Operations in specifically designated airspace. Aircraft flying in this airspace will be referred to as Autonomous Aircraft and will need to meet certain airborne equipment requirements. Autonomous Aircraft will be free to fly operator-preferred routes and their flight crews will have the task of maintaining safe separation from other Autonomous Aircraft without the intervention of Air Traffic Control. A conflict resolution methodology will be in place to enable flight crews to prevent violations of the established separation minima. To apply the conflict resolution methodology, the flight crews may need the assistance of on-board decision-support tools. Autonomous Aircraft Operations have the potential to improve the safety and efficiency of flight operations in regions of airspace with no radar-based Air Traffic Control coverage.

This thesis investigates the potential application of Distributed Artificial Intelligence concepts and techniques to develop co-operative conflict resolution methodologies for Autonomous Aircraft Operations. In this context, the term "co-operative" is used to describe conflict resolution methodologies by which conflicting Autonomous Aircraft safely co-ordinate their resolution actions so that the resolution costs are shared equitably amongst all the Autonomous Aircraft involved. A new approach to conflict resolution in Autonomous Aircraft Operations is proposed based on the sub-field of Distributed Artificial Intelligence concerned with the study of multi-agent systems. This new approach designates Autonomous Aircraft as intelligent agents and considers conflict resolution as a co-operative activity in the framework of a multi-agent system.

The means necessary for co-operation in a multi-agent system are provided by a *co-operation mechanism.* Following a review of multi-agent systems research literature, two main types of co-operation mechanisms have been identified: *behaviouristic* and *reflective.* The former type refers to co-operation mechanisms that allow the agents to be seen as acting co-operatively by an external observer, regardless of whether or not the agents are knowingly co-operating. The latter type refers to co-operation mechanisms that allow agents to engage knowingly in co-operative activity, regardless of whether or not they are seen to act co-operatively by an external observer.

To illustrate the capabilities of the proposed multi-agent approach, two examples of co-operation mechanisms that could be implemented as co-operative conflict resolution methodologies in Autonomous Aircraft Operations are presented. The first co-operation mechanism is of the behaviouristic type and has been developed specifically for an operational environment where Autonomous Aircraft can only exchange information with one another through Automatic Dependent Surveillance-Broadcast. The second co-operation mechanism is of the reflective type and has been developed specifically for an operational environment where Autonomous Aircraft can also exchange information with one another on a one-to-one basis using a point-to-point digital data-link. The two co-operation mechanisms include the necessary algorithms, protocols and procedures to design on-board decision support tools that could aid flight crews in resolving conflicts co-operatively. The performances of the co-operation mechanisms in two-dimensional conflict scenarios, involving up to three aircraft, are analysed and compared.

Acknowledgements

I wish to express my most sincere gratitude to my supervisor, Dr Colin Goodchild, for his advice, support and friendship during the years of this research.

A very special thanks to my family, especially my parents, for their continuous love, trust and support during the past 28 years.

I would also like to extend my thanks to my friends and colleagues in the ATM Research Group at Glasgow University: Stefano, Fraser, Pauline, Thierry, Kai, Hector, Leonor, Marek and Eduardo. I will keep a very warm memory of working with them.

All these years in Glasgow will be unforgettable thanks to my friends Marco, Stefano (again!), Anto, Sabrina, Xabi, Brian, Jason, Paulo, Jesus, Juan, Pauline, Abdullah, Gigs, Andrea and so many others.

I would like to dedicate this thesis to Nicola. Without her love, kindness and understanding, none of the work in this thesis would have been possible. In fact, she is the main reason why you are reading this.

Table of contents

List of figures

List of Tables

Chapter 1

Introduction

1.1 Towards a new Air Traffic Management

The purpose of Air Traffic Management (ATM) is the safe, efficient and expeditious movement of civil aircraft in the airspace [1]. The current ATM comprises of two principal ground-based services: Air Traffic Control (ATC) and Air Traffic Flow Management (ATFM). The main task of ATC is the short-term prevention of conflicts between aircraft. Conflicts are defined as violations of the established separation minima. The ATC service relies on air traffic controllers monitoring the traffic and instructing the flight crews on how to modify their routes to avoid conflicts. The ATFM service allocates air traffic flows to scarce capacity resources, both to ensure that unsafe levels of congestion do not develop and to distribute delays equitably among users. The main activity of the ATFM service is to adjust departure and arrival times to comply with airports and airspace constraints.

The major advances in the development of the concepts and procedures used to manage air traffic today were prompted by two crucial events in aviation history. The first event was the aerial campaign that took place in the skies over Europe during the Second

World War, which saw the direction of air battles and the marshalling of aircraft for the strategic bombing raids employing techniques and equipment that later developed into the current ATC system. In addition, Great Britain's vulnerability to aerial attack during that aerial campaign urged the development of an aircraft detection system based on the transmission of radio waves that was later to become a crucial tool in the hands of air traffic controllers: the radar. The second major event occurred between June 1948 and September 1949 when, in an operation known as the Berlin Airlift, over 2 millions tonnes of supplies were airlifted into West Berlin. The high air traffic density during this operation forced the introduction of new air traffic control procedures.

To cope with the expansion of commercial air traffic after the Second World War, new air traffic control procedures were developed. These new procedures relied on two types of technologies that were becoming widely available: radar and radio beacons. The idea of installing ground-based radio beacons at and between airports led to the creation of a network of air traffic routes or airways, which originated and ended at ground-based beacons. Although significant improvements have been made to date in radar and radio beacon technology, the concepts of airway and radar-based control developed during the forties are still major features in controlling air traffic today. In the current ATC, controllers on the ground still use radar to monitor aircraft flying along fixed airways and rely on voice communications to issue the appropriate instructions to the pilots so that safe separation is maintained.

As a result of the spectacular growth of air traffic during the past two decades, air traffic controllers, communication networks, airports and airspace are currently at their maximum capacity during peak times in many parts of the world. Overload and congestion, which compromise safety and cause severe economic losses to airlines, have become major issues of concern for governments, civil aviation bodies and airlines. Meanwhile, the demand for air transport continues growing and the airlines are requesting to be granted more freedom and flexibility in their operations. The ATM service as it stands today is expected to be unable to cater for the future needs of commercial aviation. The shortcomings in the current ATM were anticipated in the early eighties and since then a wide range of organisations and companies across the world have been pursuing the development of new ATM concepts based on new technologies. Some of these novel concepts are already being tested and implemented.

The new emerging ATM is expected to take advantage of advanced Communications, Navigation and Surveillance (CNS) technologies to cope with the increasing air traffic demand and provide more flexibility for airspace users while meeting adequate levels of safety.

One of the anticipated features of the future ATM is that it will allow for Autonomous Aircraft Operations (AAO) in designated regions of airspace. Autonomous Aircraft Operations will involve the transfer of the responsibility for separation assurance from the ground-based ATC to the cockpit. The aircraft flying within the regions of airspace allocated for AAO will be referred to as Autonomous Aircraft and their flight crews will exercise responsibility for separation assurance with no assistance from ground-based ATC. The concept of AAO is also associated with free routing, for it is anticipated that Autonomous Aircraft will not have to fly fixed routes. Instead, they will be allowed to fly operator-preferred routes, which could be modified dynamically without ATC clearance. The flight crew of an Autonomous Aircraft may alter its intended route to resolve conflicts with the surrounding Autonomous Aircraft, as well as to take advantage of favourable winds and avoid weather hazards. The envisaged free routing scheme in AAO has the potential to reduce fuel consumption and flight time, which would bring substantial economic benefits to airlines. Airborne separation assurance in AAO could improve the safety of flight operations in areas with no radar-based ATC coverage.

The future operational standards for AAO are currently in the process of being defined and will depend on the performance of the CNS systems available. It is anticipated that the main enabling CNS technologies for separation assurance in AAO will be the Global Navigation Satellite System (GNSS) and Automatic Dependent Surveillance–Broadcast (ADS-B). GNSS will provide an accurate and reliable positioning service available virtually at any time, any where in the world. ADS-B will provide the means for airborne surveillance by enabling aircraft to periodically broadcast their identity, position, speed and intended trajectory. In addition to GNSS and ADS-B, the necessary operational procedures, cockpit displays and decision support tools will be in place to assist the flight crews in exercising responsibility for separation assurance in a safe and efficient manner.

This thesis aims to demonstrate the potential of Distributed Artificial Intelligence (DAI) concepts and techniques to support co-operative conflict resolution in AAO. Autonomous Aircraft are deemed to resolve conflicts co-operatively when they co-ordinate their conflict resolution actions with a view not only to resolving conflicts safely but also to share the conflict resolution costs fairly. The conflict resolution costs are related to Autonomous Aircraft having to deviate from their operator-preferred route to resolve conflicts. This thesis will propose two co-operative conflict resolution methodologies for AAO based on concepts and techniques from DAI. Each of these two methodologies corresponds to a different possible future operational environment for AAO. The two methodologies include procedures, protocols and algorithms inspired by research in the field of multi-agent systems, a sub-field of DAI. The flight crews of Autonomous Aircraft will have the necessary decision-support tools at their disposal to assist them in applying the proposed methodologies.

In the remainder of this section the main institutional efforts towards the implementation of a new ATM are reviewed. The work of the International Civil Aviation Organisation (ICAO) to establish an efficient ATM on a global level is described in subsection 1.1.1. Subsequently, the main new concepts and procedures for the future ATM proposed by the United States Federal Aviation Administration (FAA) and the European Organisation for the Safety of Air Navigation (Eurocontrol) are outlined in subsections 1.1.2 and 1.1.3, respectively. The rest of this chapter is structured as follows. In section 1.2 the Airborne Separation Assurance System (ASAS) is described. ASAS embodies ICAO's approach to the issue of the transfer of responsibility for separation assurance from the ground-based ATC to the flight crew under the future ATM. At the end of the section, Autonomous Aircraft Operations are explained in the context of the envisaged ASAS applications. Section 1.3 introduces the operational concept for Autonomous Aircraft Operations that is proposed as the subject of study of this thesis. In this operational concept, conflict resolution in AAO is seen as an essentially co-operative activity. Section 1.4 contains a review of previous research relating to Autonomous Aircraft Operations. The objectives of the research described in this thesis, as well as its main contributions are summarised in section 1.5. This chapter concludes with section 1.6, which outlines the remaining chapters of the thesis.

1.1.1 The International Civil Aviation Organisation (ICAO)

In 1983 the International Civil Aviation Organisation (ICAO), which is the United Nations agency regulating international air transport, established the Special Committee on Future Air Navigation Systems (FANS Committee). The FANS Committee considered the steady growth of air transport preceding 1983 and identified the shortcomings inherent in the communications, navigation and surveillance systems, and in the operational procedures supporting civil aviation at the time. The Committee determined that those systems and procedures were incapable of coping with the future needs of international air transport. In its final report presented to the President of the ICAO Council in 1988, the FANS Committee exposed the need to develop new systems and procedures that overcome the limitations of ATM and allow it to evolve on a global scale. The Committee also recognised that the final achievement of a world-wide ATM system would require sovereign nations to change the way in which they deal with the implementation of civil aviation systems. Thus, they would have to make a compromise between their political and military interests and the international air transport needs.

In 1989 ICAO created the Special Committee for the Monitoring and Co-ordination of Development and Transition Planning for the Future Air Navigation Systems (FANS Phase II Committee), which would continue the work of the FANS Committee. The FANS II Committee determined that the goals of the future global ATM should include the enhancement of safety, the provision for a more flexible and efficient use of airspace and the creation of a homogeneous global airspace. The Committee finished its work in 1993 and by that time the FANS concept had become known as Communications, Navigation, Surveillance/Air Traffic Management systems (CNS/ATM systems). The concept of CNS/ATM systems involved a complex set of existing and emerging inter-related technologies, which were expected to enhance the performance of the existing air traffic management practises around the world by enabling a global ATM.

As a result of the conclusions and recommendations of the two FANS Committees, ICAO initiated the development of a plan for a global implementation of the CNS/ATM systems. This development culminated with the presentation of the Global Air

Navigation Plan for CNS/ATM Systems [2] to the World-wide CNS/ATM Systems Implementation Conference, held in Rio de Janeiro in 1998. This Global Plan describes ICAO's approach to the implementation of CNS/ATM on the global, regional and national levels, with the aim of unifying diverse local needs and forming them into a coherent strategy. The Global Plan also presents a broad ATM operational concept developed by ICAO, which reflects the latest information available on CNS/ATM at the time. The Global Plan was developed as an evolving document comprising technical, operational, economic, financial, legal and institutional elements, and offering practical guidance and advice to regional planning groups and individual countries on implementation and funding strategies.

ICAO circulated detailed air traffic forecasts to support the implementation of the Global Plan (see Table 1.1). The projected growth of international air transport between 1995 and 2005 shown in the forecasts fostered the international commitment to implement the CNS/ATM systems concept.

TOTAL SCHEDULED SERVICES					
				Average annual growth rate (%)	
	Actual 1985	Actual 1995	Forecast 2005	1985-1995	1995-2005
Total number of passengers (10^6)	899	1285	2010	3.6	4.5
Total Passenger-kilometres (10^9)	1367	2228	3807	5.0	5.5
Passenger-kilometres (10^9) by region of airline registration					
Africa	36.7	51.0	77	3.3	4.0
Asia-Pacific	222.3	549.7	1260	9.5	8.5
Europe	428.2	549.3	870	2.5	4.5
Middle East	42.7	67.0	115	4.6	5.5
North America	569.2	902.7	1310	4.7	4.0
Latin America and Caribbean	68.3	107.9	175	4.7	5.0

Table 1.1: Summary of the ICAO Air Traffic Forecasts for the year 2005 [3].

According to the Global Plan, aeronautical communications will increasingly take place via digital data-link, whilst satellite voice and data communications providing global coverage will be added to the existing communications channels. An Aeronautical Telecommunications Network (ATN) is expected to support the exchange of digital data over air-air, air-ground and ground-ground interconnected sub-networks. The implementation of a global ATN will radically improve the current level of information sharing in ATM.

Regarding navigation, ICAO's Global Plan considers the progressive introduction GNSS together with the widespread use of Area Navigation (RNAV), which releases aircraft from having to fly along fixed airways referenced to ground navigation aids. These projected improvements are expected to provide global navigation support and allow airlines to fly more efficient routes.

The Global Plan predicts a major breakthrough in surveillance with the implementation of Automatic Dependent Surveillance (ADS). ADS enables aircraft to automatically transmit their position and other additional information contained in the Flight Management System (FMS) to the ATC, via satellite or other communication channels. The concept of ADS will also be applied in airborne surveillance using ADS-B. In addition to ADS and ADS-B, conventional Secondary Surveillance Radar (SSR) modes will continue to be extensively used, along with the gradual introduction of Mode S.

Figure 1.1 depicts an overview of the expected benefits that the new ATM will deliver to the civil aviation community according to ICAO's Global Plan. In addition to these general benefits, the new approach to communications, navigation and surveillance proposed by ICAO is expected to be able to foster the growth of air transport in developing regions [4]. While many developing countries cannot afford the implementation and maintenance of existing ground-based ATM technology, the ICAO's CNS/ATM systems concept could be achieved in those countries by taking advantage of shared investment by service providers and airlines. ADS could make low cost air traffic control centres affordable and GNSS, ADS-B and air-to-air data-link communications could make safe navigation and airborne-based separation assurance possible in remote areas where the installation of radar and ground control centres is impossible or unaffordable.

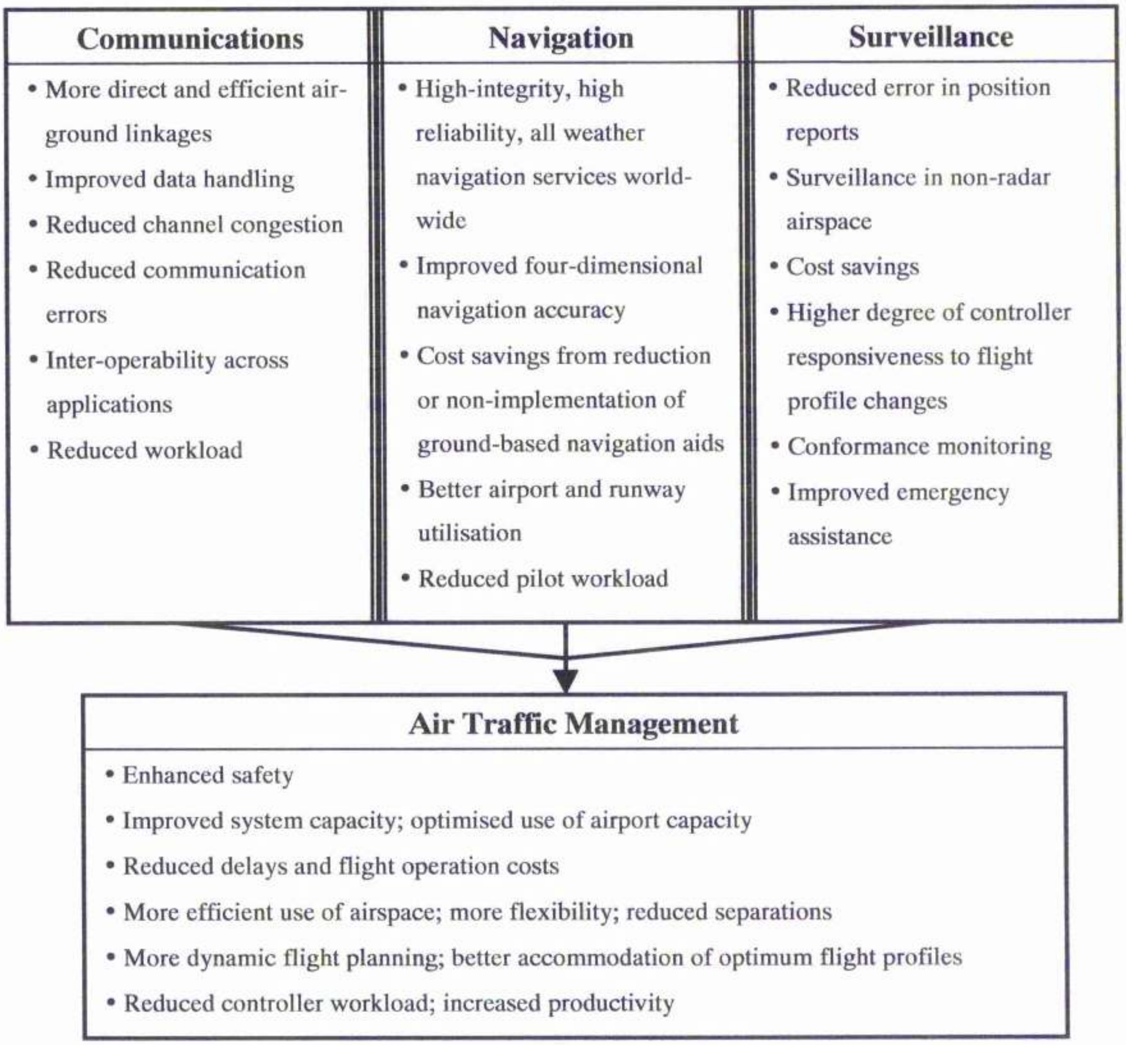


Figure 1.1: Overview of the expected benefits of ICAO CNS/ATM systems concept [2].

1.1.2 The Federal Aviation Administration (FAA)

The National Airspace System (NAS) comprises of the entire civil aviation infrastructure in the United States. The Federal Aviation Administration (FAA) is the United States governmental organisation in charge of managing the NAS. The FAA approach to the ICAO CNS/ATM systems concept is embodied in the concept of *Free Flight*. In 1995, following the advice of the Radio Technical Commission for Aeronautics (RTCA), which is a United States-based not-for-profit private corporation that addresses requirements and technical concepts for aviation, the FAA endorsed Free

Flight as its guiding concept and future ATM operational framework. Earlier that year, the RTCA had been requested by the FAA to create a Free Flight Implementation Task Force. In the Final Report elaborated by the Task Force, Free Flight is defined as:

> "... a safe and efficient flight operating capability under instrument flight rules (IFR) in which the operators have the freedom to select their path and speed in real time. Air traffic restrictions are only imposed to ensure separation, to preclude exceeding airport capacity, to prevent unauthorised flight through Special Use Airspace (SUA), and to ensure safety of flight. Any activity which removes restrictions represents a move toward Free Flight." [5].

In that Final Report, the RTCA proposed an incremental evolution towards the implementation of Free Flight (see Figure 1.2).

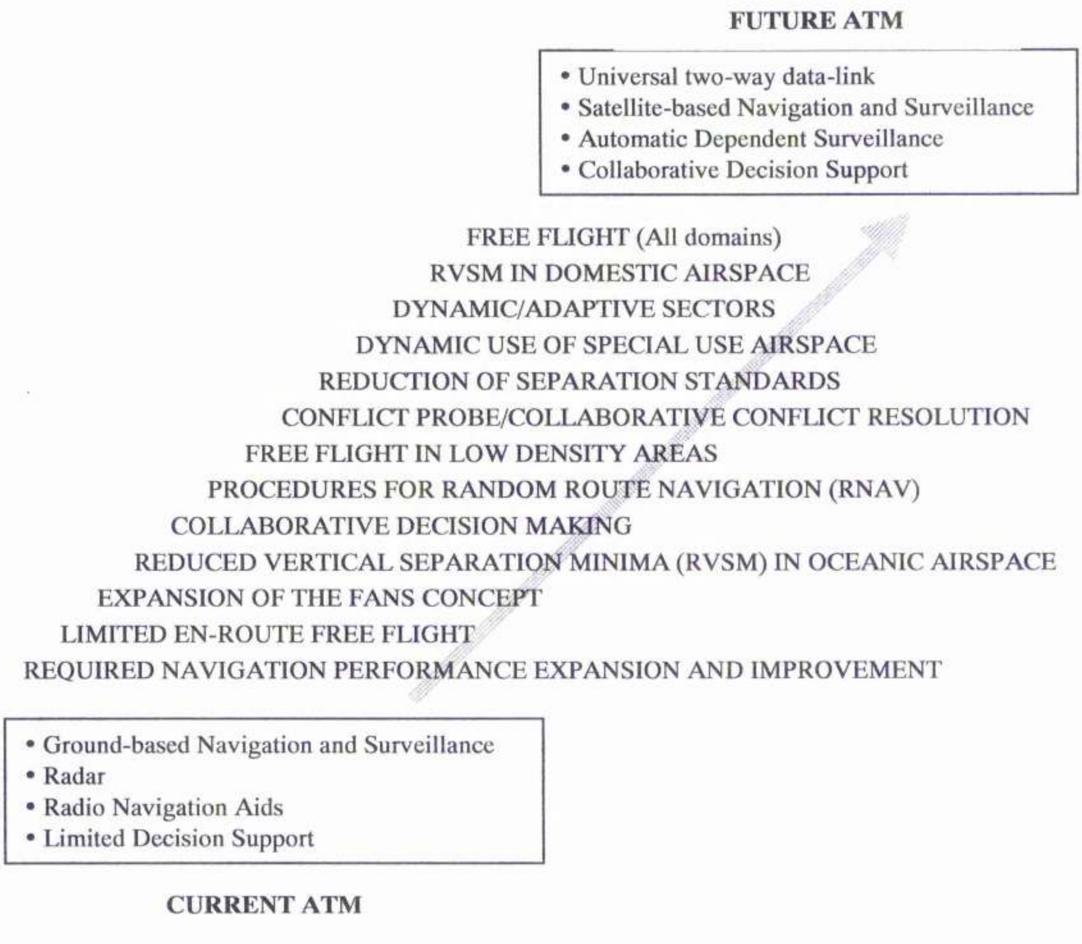


Figure 1.2: Free Flight and the path to the future ATM according to RTCA [5].

In mature Free Flight operations, communication, navigation and surveillance technologies together with new ATM procedures would make an increase in the airspace capacity possible by reducing separation standards and would allow the airspace users to fly their preferred routes while guaranteeing adequate levels of safety. To achieve these objectives, Free Fight is anticipated to rely on extensive dynamic collaboration between the airspace users and all the establishments involved in ATM.

Depending on the traffic density and the complexity of the traffic flow, Free Flight will range from total route freedom and flexibility to four-dimensional flight plan contracts. Within Free Flight, it is anticipated that, in agreed and appropriate circumstances, the task of maintaining safe separation between aircraft could be shared between ATC and the flight crew or even fully transferred to the cockpit.

In 1998 the FAA launched Free Flight Phase 1 (FFP1) as the first step in the evolutionary process towards Free Flight [6]. The aim of FFP1 is the limited deployment of a set of new systems and operational tools to evaluate their performance and the early benefits achieved as a result of their implementation. These new capabilities are expected to be available for a more widespread deployment by the end of the program in December 2002.

The five new capabilities being assessed within FFP1 are outlined below:

- **Collaborative Decision Making (CDM)**, which provides airlines and the FAA with real-time access to NAS-related information such as weather and delays. The aim of CDM is to foster collaboration between airspace users and air traffic managers to achieve more efficient utilisation of the airspace.

- The **User Request Evaluation Tool (URET)**, which enables controllers to manage pilot requests in en-route airspace by identifying potential conflicts up to 20 minutes ahead.

- **The Traffic Management Advisor**, which provides computer automation to support arrival sequence planning in the extended terminal airspace surrounding major airports in the United States. TMA is expected to increase the operational efficiency in the airspace between en-route flight and the final approach.

- The **passive Final Approach Spacing Tool (pFAST)**, which aims to maximise arrival throughput by providing the controller with optimal aircraft landing sequences and runway allocations.

- The **Surface Movement Advisor (SMA)**, which provides airlines with aircraft arrival information to enhance gate and ramp operations and reduce taxi delays.

In addition to the ground-based capabilities being evaluated by FFP1, the FAA is also involved in the operational evaluation of new cockpit-based tools and procedures within the program Safe Flight 21 [7]. Safe Flight 21 is an FAA and industry collaborative project aiming to assess operational enhancements that address the needs of the aviation industry and contribute towards the implementation of the Free Flight concept. The program comprises of a series of flight trials to be performed between 1999 and 2002. The aim of these trials is to demonstrate and validate new operational enhancements in a real-world environment. Some of these enhancements are listed below:

- Weather and other Information in the cockpit.
- Cost-effective avoidance of Controlled Flight into Terrain (CFIT).
- Improved terminal operations in low visibility.
- Enhanced see and avoid.
- Improved surface navigation for the pilot.
- Enhanced surface surveillance for the controller.
- ADS-B surveillance in non-radar airspace.

The main enabling technologies for these operational enhancements are ADS-B and Traffic Information Service-Broadcast (TIS-B), which allows for traffic and other data available on the ground to be transmitted to the cockpit via a broadcast-mode data-link. ADS-B and TIS-B information would be displayed to the pilot on an advanced multifunctional display, the Cockpit Display for Traffic Information (CDTI).

1.1.3 The European Organisation for the Safety of Air Navigation (Eurocontrol)

Eurocontrol was founded in 1960 by six European states with the mission of overseeing air traffic control in their upper airspace. Currently Eurocontrol has a membership of twenty-nine states and its main objective is to organise co-operation in ATM between the respective national administrations [8]. Eurocontrol's importance grew as it implemented and managed the strategy of the European Civil Aviation Conference for the 1990's.

The European Civil Aviation Conference (ECAC) was established in 1955 by nineteen European states with the active support of ICAO. ECAC is an inter-governmental organisation with the objective of promoting the continued development of a safe, efficient and sustainable European air transport system [9]. Currently ECAC comprises of thirty-eight member states and seeks to harmonise their civil aviation practices and to promote understanding in aviation policy matters between the member states and other parts of the world.

One of Eurocontrol's major achievements was to establish the Central Flow Management Unit (CFMU) in Brussels in response to an initiative from ECAC. In addition, Eurocontrol has initiated programmes to optimise, harmonise and integrate air traffic control at centres and airports across Europe [8]. Most recently, Eurocontrol has developed the ATM Strategy for the years 2000+ [10], which lays down the framework for the necessary improvements in airspace organisation, infrastructure and procedures to meet the projected demand for air traffic in Europe between 2000 and 2015. The ATM Strategy for 2000+ was developed at the request of the ECAC member states' Ministers of Transport, who endorsed it in January 2000. The Strategy emphasises the need to create a single airspace for Europe, which, for ATM purposes, shall not be constrained by national boundaries [10]. The main innovations proposed by Eurocontrol to realise the objectives of the ATM Strategy for 2000+ are presented in the Operational Concept Document (OCD) [11]. The OCD provides a high-level description of the target operational concept for the European ATM in 2015.

According to the OCD, the ECAC airspace will comprise the three types of airspace described below:

- **Unmanaged Airspace (UMAS)**, which will correspond to the airspace currently referred to as non-controlled airspace. UMAS will be subject to the same Rules of the Air [12] applied today in non-controlled airspace. Aircraft operating in UMAS will be provided with traffic-related information on request.

- **Managed Airspace (MAS)**, in which the ground-based ATM provider will be responsible for separation assurance. MAS will comprise of the volumes of airspace around airfields, known as Terminal Manoeuvring Areas (TMAs), as well as volumes of en-route airspace. In areas within MAS with a low density of air traffic, aircraft will be allowed to operate user-preferred routes, while in busy areas the traffic will be organised in the form of a route network that could change dynamically to maximise capacity and efficiency.

- **Free Flight Airspace (FFAS)**, in which aircraft will be allowed to fly operator-preferred routes and the separation assurance related tasks will be transferred to the flight crew. Aircraft operating in FFAS will be free to dynamically modify their intended trajectories. Although the responsibility for separation assurance will rest on the flight crew, the ground-based ATM provider could, in principle, intervene in non-nominal situations. FFAS will be allocated on a daily basis according to the expected traffic demand by an airspace planning service within ATM. In principle, access to FFAS will only be granted to suitably equipped aircraft.

Figure 1.3 depicts a schematic plan view of the envisaged ECAC airspace structure for 2015.

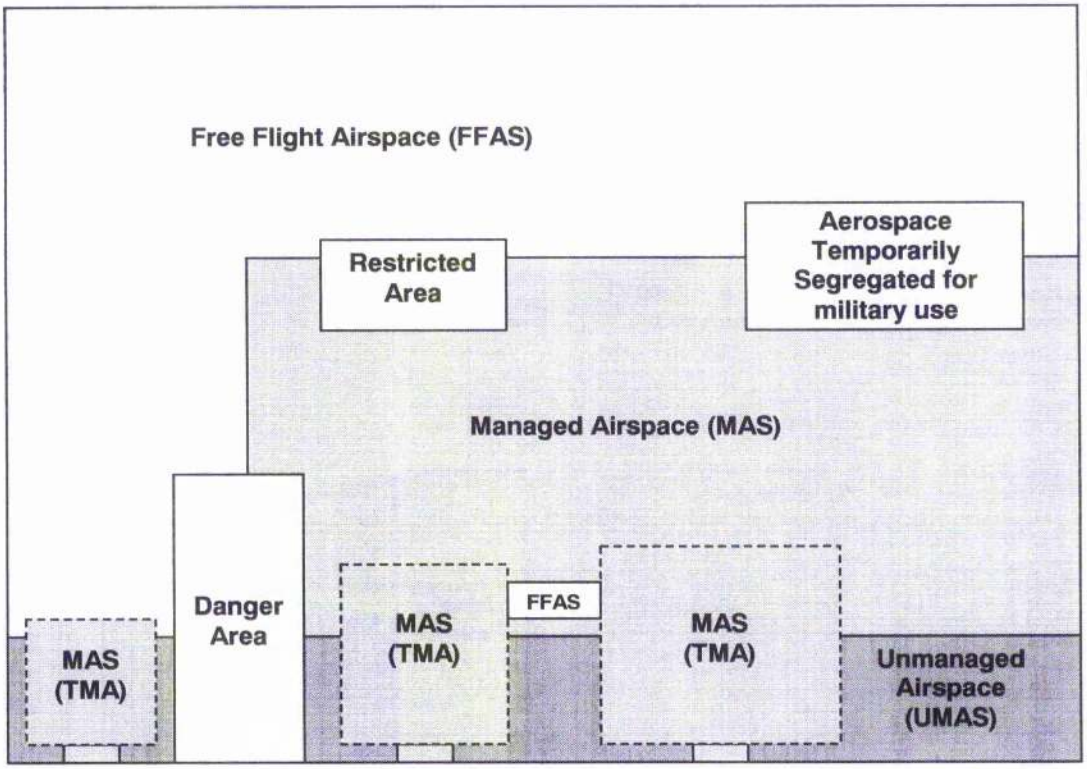


Figure 1.3: Vertical view of the predicted European airspace structure for 2015 [11].

1.2 The Airborne Separation Assurance System (ASAS)

The ASAS concept is the response of ICAO to the diverse emerging tools and procedures being developed to support a further engagement of the flight crew in separation assurance under the future ATM. With the development of the ASAS concept, ICAO aims to define international standards that will regulate and harmonise those new tools and procedures. In this section, the ASAS concept will be defined and its anticipated applications described. The objectives and contributions of the work described in this thesis will be explained in the context of one of the ASAS applications: Autonomous Aircraft Operations.

1.2.1 The ASAS concept

The ASAS concept was first introduced in 1995 by the ICAO Secondary Surveillance Radar Improvements and Collision Avoidance Systems Panel (SICAS Panel) [13], which defined it as:

> "The equipment, protocols and other aircraft state data, flight crew and ATC procedures which enable the pilot to exercise responsibility, in agreed and appropriate circumstances, for separation of his aircraft from one or more aircraft" [14]

The ASAS concept comprises of two broad categories of proposed applications [15], which are introduced below:

- **Traffic Situational Awareness Applications**, which are related to the provision of information to the flight crew regarding position, identity, flight status and intentions of proximate aircraft.

- **Co-operative Separation Applications**, where the pilot uses ASAS equipment to perform operational procedures that aim to maintain defined separation minima with proximate aircraft.

Although the ASAS applications are still in the research and development stage, they are seen as cornerstones of the future ATM. The main anticipated ASAS applications within the two main categories presented above are examined in detail in subsection 1.2.3. The FAA-Eurocontrol Research and Development Committee has recently issued a document entitled "Principles of Operations for the Use of ASAS" [16], in which a more exhaustive categorisation of the anticipated ASAS applications is proposed. This document aims to harmonise ASAS research in Europe and the USA.

The main benefits expected to be delivered by the ASAS applications have been anticipated by the SICAS Panel [13] and are outlined below:

- Improvement of the pilot's situational awareness. Operational safety is expected to improve with the provision of information regarding identity, status, position and intentions of the proximate aircraft.

- Increase in the capacity and improvement of the efficiency of ATC through the active involvement of the aircraft crew in the separation assurance process. The delegation of the responsibility for separation assurance to the cockpit is expected to reduce the controllers' workload.

- Increase in the airspace capacity by enabling a more accurate compliance to separation minima. Ultimately, ASAS is expected to contribute to the reduction of these separation minima.

The SICAS Panel maintains that ASAS should be kept independent from the Airborne Collision Avoidance System (ACAS) ([17], [15], [14]), which is the standard on-board system to prevent imminent mid-air collisions when the primary means for separation fails. ACAS provides the pilot with traffic situation awareness, traffic proximity warning, imminent collision alert and recommended collision avoidance manoeuvres. To do so, ACAS tracks the proximate aircraft by interrogating their Secondary Surveillance Radar (SSR) transponders. The operation of ACAS is independent of the aircraft navigation equipment and the ground-based ATM. A high risk of collision triggering an ACAS alert indicates a malfunction of the primary means of separation assurance. Therefore, ACAS has to remain independent from this means, regardless of where the responsibility for separation assurance is placed. ACAS is described in detail in Appendix A.

ASAS applications will have to be compatible with ACAS with regards to operational procedures [18]. The relationship between ASAS and ACAS will have to be carefully defined prior to the implementation of any ASAS application. In principle, ACAS and ASAS might share some components, but this must not be detrimental to the ACAS function [18]. ASAS and ACAS could even collaborate to enhance global operational safety. In fact, the work described in [19] recommends the use of only one CDTI to present both ACAS and ASAS information and suggests that the exchange of data between ACAS and ASAS could improve the ASAS surveillance function and enable the compatibility of ASAS and ACAS alerts.

1.2.2 Enabling technologies for ASAS

1.2.2.1 Automatic Dependent Surveillance-Broadcast (ADS-B)

ASAS applications are expected to rely on information provided by an onboard surveillance system such as Automatic Dependent Surveillance-Broadcast (ADS-B). According to the ICAO Automatic Dependent Surveillance Panel, ADS-B can be defined as a function that enables aircraft to periodically broadcast their state vector, which contains the aircraft position and velocity, together with other information [20]. ADS-B is "automatic" because no external stimulus is required to trigger a transmission. It is "dependent" because the surveillance information transmitted is dependent on and derived from the aircraft's on-board systems. In principle, any user within the surveillance range, either aircraft or ground-based, may use and process ADS-B surveillance information.

In 1998, the RTCA issued a document containing the minimum performance requirements for ADS-B [21]. However, ADS-B is still in the development stage and there is no international agreement on which data-link technology will finally enable the broadcasting of ADS-B data. Currently, three different data-link technologies presumably capable of supporting the ADS-B function are being examined: Mode-S Extended Squitter, VHF Data Link Mode 4 (VDL Mode 4) and Universal Access Transceiver (UAT). The level of operational performance of the ADS-B function will depend on the final characteristics and capabilities of the chosen data-link. Regardless of the favoured data-link, the ADS-B function will have to comply with international minimum operational standards. Thus, it is possible that more than one data-link technology will be approved to support ADS-B as long as the ADS-B function displays the appropriate operational performance levels. The three data-link technologies are described in detail in Appendix B.

1.2.2.2 Air-to-air data-link

Advanced ASAS applications may also require the use of an air-to-air data-link to enable aircraft to address specific aircraft in their vicinity. In addition to the broadcasting ADS-B function, this data-link capability would allow aircraft to exchange data with selected proximate aircraft on a one-to-one basis. This high-performance inter-aircraft data communications capability is anticipated to make the co-ordination of separation assurance manoeuvres between aircraft possible [22]. This service is still in the research phase and its technical and operational requirements have not been established.

1.2.2.3 Cockpit Display of Traffic Information (CDTI)

A crucial issue concerning the implementation of ASAS applications is the satisfactory interaction between the pilot and the ASAS equipment. It is anticipated that the Cockpit Display of Traffic Information (CDTI) will be an essential component of the interface between the pilot and the ASAS applications. The CDTI will display to the pilot the identities and relative positions of the proximate aircraft based on ADS-B surveillance data. Other data received through ADS-B, such as aircraft intent, weather data and information received from the ground-based ATM service will also be presented to the pilot in the CDTI, either automatically or by pilot's request. The CDTI is expected to be the core element of traffic situational awareness ASAS applications. The ASAS co-operative separation applications are also expected to rely on the use of the CDTI.

1.2.2.4 Traffic Information System-Broadcast (TIS-B)

It is expected that mixed-equipage traffic situations involving aircraft with and without ADS-B will occur in the future. In these situations, the information relating to the proximate traffic will be based on both ADS-B surveillance data and radar data uploaded from a ground-based station through TIS-B (Traffic Information System-Broadcast). TIS-B consists of the broadcasting of radar information via data-link, to provide aircraft with surveillance information concerning their proximate traffic. TIS-B will complete the picture regarding the traffic surrounding an aircraft by providing surveillance data for the proximate aircraft not equipped with ADS-B.

1.2.3 ASAS applications

1.2.3.1 Traffic Situational Awareness applications

The aim of the Traffic Situational Awareness (TSA) applications is to provide the pilot with an accurate picture of the surrounding traffic. The provision of TSA does not involve the transfer of responsibility for separation assurance from the ground-based ATC to the cockpit. It is expected that TSA will be accomplished by displaying ADS-B and TIS-B data on the CDTI. The use of the CDTI for separation purposes shall be identified as a Co-operative ASAS application.

1.2.3.2 Co-operative ASAS applications

In the current ATC system, pilots are in charge of the efficient navigation and control of their aircraft, whereas air traffic controllers are responsible for maintaining aircraft separation. Thus, in controlled airspace pilots have to follow controllers' directions to ensure safe separation from proximate aircraft. Controllers issue instructions to pilots to comply with separation minima and therefore achieve safe and efficient air traffic operations. Pilots themselves have no separation minima to maintain, other than avoid imminent collisions and wake turbulence [22]. However, co-operative ASAS applications will transfer separation assurance tasks to the flight crew. Under determined circumstances and providing that the adequate tools and procedures are in place, the flight crew will exercise responsibility for complying with an ATC clearance that involves maintaining safe separation from certain other aircraft. The flight crew will co-operate with ATC to preserve safe separation between its aircraft and the surrounding traffic. This implies a new share of responsibilities between the ground-based ATC and the flight crew, which needs to be clearly defined due to its legal implications. The Review of the General Concept of Separation (RGCS) Panel of ICAO distinguishes between two levels of transfer of responsibility for co-operative ASAS applications [22]:

- **Limited transfer of responsibility:** ATC remains responsible for separation assurance, except in determined circumstances defined in a period of time, a volume of airspace and a level of complexity of traffic. In such circumstances, the flight crew could assume the responsibility for separation assurance within the boundaries of an ATC clearance. ATC would establish which separation assurance tasks are delegated to the flight crew. Limited transfer of responsibility is anticipated to make an increase in ATC capacity possible through a reduction of the controllers' workload, an accurate compliance with the separation minima and a possible reduction of those minima. An example of a co-operative separation application involving Limited Delegation is a station keeping application in which an aircraft flying behind another one is cleared by the controller to follow the leading aircraft and maintain a certain longitudinal separation from it.

- **Extended transfer of responsibility:** The responsibility for separation assurance is fully assumed by the flight crew. The ground ATM authority would only be responsible for monitoring the traffic complexity and maintaining it at a level compatible with the airborne separation assurance capabilities. Autonomous Aircraft Operations is an example of an ASAS application involving extended transfer of responsibility.

In addition to distinguishing between two levels of transfer of responsibility, the ICAO RGCS Panel defines three different levels of *delegation of separation assurance* in co-operative separation applications. The delegation of separation assurance involves the assignation of specific separation assurance tasks to the flight crew. The RGCS Panel describes the separation assurance process as consisting of four consecutive tasks, regardless of where the responsibility for the process is placed [22]:

- **Task 1- Conflict detection**, which involves the analysis of the traffic situation and the detection of possible violations of the established separation minima between the aircraft considered. Conflict detection can be performed either by a human operator or by an automated conflict detection tool.

- **Task 2- Determination of a conflict resolution strategy**, possibly with the assistance of automated conflict resolution tools.

- **Task 3- Implementation of the conflict resolution strategy**, which involves the flight crew manoeuvring the aircraft according to the strategy formulated to resolve the conflicts.

- **Task 4- Monitoring of the conflict resolution strategy**, which involves a human operator observing the aircraft trajectory to establish that the conflict resolution manoeuvre achieves its objective. If the objective is anticipated to be unattainable through the current manoeuvre, the separation assurance process restarts in task 1.

Depending on which of these tasks are assigned to the cockpit during the separation assurance process, the three levels of delegation as noted by the RGCS Panel are as follows [22]:

- **Limited delegation:** Tasks 1 and 2 (conflict detection and determination of the strategy) are performed by ATC. Tasks 3 and 4 (implementation and monitoring) are allocated to the flight crew.

- **Extended delegation:** Task 1 is performed by ATC. Tasks 2, 3 and 4 are assigned to the cockpit.

- **Full delegation:** Tasks 1, 2, 3 and 4 are assigned to the cockpit.

Limited delegation and extended delegation are associated with limited transfer of responsibility for separation assurance, while full delegation of separation requires an extended transfer of responsibility for separation assurance. The level of transfer of responsibility, the level of delegation of separation and the relationship between them have to be clearly defined for each future ASAS application. Thus, ASAS standard operational procedures will have to be in place and the roles and responsibilities of pilots and controllers will have to be precisely determined. The capabilities of the CNS airborne equipment will have to be considered to define the ASAS standard operational procedures. They will also have to be taken into account in the design of possible decision support tools for the pilot and in the definition of the separation minima for the different ASAS applications.

1.2.4 Autonomous Aircraft Operations (AAO)

The term "Autonomous Aircraft Operations" refers to a future operational concept involving the full transfer of both separation assurance and trajectory management functions to the flight deck in low density airspace ([23], [24]). The future ATM is envisaged to support Autonomous Aircraft Operations in specifically designated regions of airspace. In Eurocontrol's Operational Concept Document for the future European ATM [11], the term Free Flight Airspace (FFAS) is introduced, referring to the regions of airspace allocated for Autonomous Aircraft Operations. FFAS is anticipated to be available principally in regions of the European upper airspace with no radar coverage. Autonomous Aircraft Operations could also be implemented in regions of airspace elsewhere, chiefly over oceanic and remote continental areas with no radar-based ATC coverage. The term AAO airspace will be used in this thesis to refer to any region of airspace specifically allocated for Autonomous Aircraft Operations. The aircraft flying within AAO airspace will be referred to as Autonomous Aircraft.

The flight crews of Autonomous Aircraft will exercise responsibility for maintaining safe separation from one another with no ATC assistance. Thus, a co-operative ASAS application involving extended transfer of responsibility for separation assurance and full delegation of the separation assurance will have to be implemented to support AAO. In addition, Autonomous Aircraft will be allowed to fly operator-preferred routes, modifying their filed flight plan at any time without ATC clearance. Allowing flight crews the freedom to select and dynamically adjust their routes within AAO airspace could bring substantial benefits to airlines.

Autonomous Aircraft Operations could improve the safety and efficiency of flight operations in regions of airspace where radar-based ATC is not available, for example, over the oceans and remote continental areas. Currently, monitoring and control of air traffic in this type of airspace is based on flight plan data, position estimates, and relayed voice position reports from the pilots. Fixed route structures are used to impose order and separate the traffic flows. Separation assurance is achieved through procedural ATC. Under procedural ATC, the controller issues clearances to the pilots to fly along specified routes through the procedurally controlled region of airspace. The pilots must comply with their assigned routes to ensure safe separation. Additionally, controllers monitor flight progress based on periodical pilot position reports. With each report, the controller re-examines the traffic situation and searches for potential conflicts. Procedural ATC requires large lateral and longitudinal separation minima between aircraft to take into account uncertainties, lack of timely surveillance and delays associated with the communications. The introduction of AAO in airspace regions currently under procedural ATC would provide a means for flight crews to be made aware of their surrounding traffic and able to detect and resolve conflicts safely and efficiently with no ATC assistance. This would open the way for a reduction of the current separation minima in these regions of airspace. In addition, the removal of the requirement to fly along fixed routes would free up airspace currently unused by aircraft flying under procedural control.

The implementation of AAO in regions of airspace currently under radar-based ATC could allow for an increase in the efficiency and flexibility of airlines operations in those regions of airspace. From an ATC perspective, the fact that the fight crews of Autonomous Aircraft will hold full responsibility for separation assurance could contribute to alleviating controllers' workload in certain sectors.

The operational and technical standards for Autonomous Aircraft Operations are yet to be defined. Consequently, an operational concept for AAO has been proposed as the subject of study of this thesis. The main feature of this concept, which is described in section 1.3 below, is that it involves co-operative conflict resolution[1]. The proposed operational concept includes a definition of the minimum requirements for a future operational environment where that concept could be implemented. An operational environment for the proposed concept of AAO would include the necessary airborne CNS equipment, the ground-based services available and the airspace structure in place.

[1] In this context, the term "co-operative" does not refer to "co-operative ASAS applications". Instead, as in section 1.1, it is used to indicate that conflict resolution in AAO involves the Autonomous Aircraft co-ordinating their conflict resolution actions with a view to equitably sharing the resolution costs.

Given a specific operational environment, a conflict resolution methodology compatible with that environment would have to be designed to facilitate the implementation of the proposed concept for AAO. The conflict resolution methodology would comprise of the protocols, procedures and algorithms necessary to enable flight crews to resolve conflicts according to the proposed concept for AAO in that operational environment. This thesis is devoted to investigate the potential application of ideas and techniques associated with the field of Distributed Artificial Intelligence (DAI) to design conflict resolution methodologies that could support the proposed concept of AAO in two different operational environments.

1.3 Operational concept for Autonomous Aircraft Operations with co-operative conflict resolution

Once an aircraft enters AAO airspace it becomes an Autonomous Aircraft and is allowed to fly its operator-preferred route. ATC delegates the four separation assurance tasks described above in section 1.2. The flight crew holds extended transfer of responsibility for separation assurance. Within AAO airspace ASAS is the sole means for separation assurance and ACAS will be in place as a safety net.

1.3.1 Ground-based ATM services for AAO

The role of ATC in AAO is assumed to be limited to managing the transfer of aircraft from AAO airspace to ground-based controlled airspace and vice versa. It is anticipated that controllers and flight crews will collaborate in ensuring the safety and efficiency of the flight while entering and exiting AAO airspace. The development of new decision-support tools and operational procedures to enable them to do so is one of the key issues in the implementation of AAO. This thesis will focus on flight operations once the aircraft are within AAO airspace and are fully responsible for separation assurance. Issues relating to the transfer to and from ground-controlled airspace are considered out of the scope of this work. An outline of the research on transfer issues performed at Glasgow University can be found in [25].

It is assumed that a new ground-based ATM will periodically re-allocate the AAO airspace regions according to Air Traffic Flow Management constraints. The aim of this ATM service is to ensure that Autonomous Aircraft Operations are confined to regions of airspace in which the air traffic density and complexity are such that potential conflicts are deemed to be manageable by the Autonomous Aircraft.

1.3.2 Minimum airborne equipment requirements for AAO

To operate within a volume of airspace designated for Autonomous Aircraft Operations, aircraft must be fitted with certain CNS and ASAS equipment to ensure that they are capable of performing airborne separation assurance safely. In this thesis it is assumed that any operational environment for AAO must include the following minimum airborne equipment requirements for Autonomous Aircraft:

- **GNSS**, which provides the aircraft with an accurate positioning service as well as with a reliable and precise common time reference.

- **Four-dimensions Flight Management System (4D FMS)**, which enables aircraft to accurately fly their intended trajectories. The 4D FMS allows the aircraft to conform to four-dimensional flight plans, which describe the aircraft's intended route in terms of a sequence of three-dimensional positions that are to be reached within an established time frame [26].

- **ADS-B**, which allows aircraft to transmit their identity, status, position, velocity and intended route to the surrounding aircraft.

- **ASAS equipment**, which assists the flight crew in detecting and resolving conflicts. The ASAS equipment compares the aircraft's intended route with its proximate aircraft's predicted future trajectories to detect possible conflicts. When a conflict is detected, it assists the pilot in the resolution process according to the conflict resolution methodology in place.

- **CDTI**, which displays the information received through ADS-B and provides the pilot with an accurate picture of proximate traffic. The CDTI could also be viewed as part of the ASAS equipment.

- **ACAS**, which is the last recourse to prevent mid-air collisions when ASAS fails to ensure safe separation. ACAS acts as a safety net and is assumed to function independently from ASAS.

In addition to the minimum equipment requirements outlined above, Autonomous Aircraft may also be able to communicate with one another using a point-to-point data-link. This data-link would allow for the exchange of information between aircraft on a one-to-one basis and could facilitate the co-ordination of conflict resolution actions.

1.3.3 Airborne separation assurance in AAO

Currently, two aircraft in controlled airspace are considered to be safely separated when there is no immediate danger of them violating the established separation minima. A violation of the separation minima is called a conflict and involves an unacceptable risk of collision. Controllers issue instructions that enable flight crews to maintain the established separation minima with the surrounding aircraft. Aircraft are separated by at least the established separation minima to keep the risk of them colliding with each other below an acceptable value. This value is called the Target Level of Safety (TLS) and depends on the rate of fatalities due to airborne collisions that the public is prepared to accept.

In the current international regulations, longitudinal, lateral and height separation minima are defined for a pair of aircraft in terms of criteria based on distance, geometry or time, depending on the circumstances. Air traffic controllers can ensure safe separation by enforcing either lateral, longitudinal or vertical separation minima. Under radar control, the separation minima depend on factors such as the type of flight operations (en-route or TMA), the aircraft's altitude and the radar surveillance performance. The radar separation minima currently applied range from 2.5 to 10 nautical miles (nm). Vertical separation is obtained by requiring aircraft to fly at different flight levels, which are set at 1,000 or 2,000 feet (ft) intervals, depending on the altitude and type of airspace. Outside of radar coverage, the minimum lateral spacing between fixed routes is typically 10 nm. An aircraft operating in one of these routes must maintain an established longitudinal time distance with the aircraft flying in front of it along the same route. This minimum longitudinal time separation is commonly 10 minutes.

A definition of safe separation between aircraft for the proposed concept of AAO is presented next. This definition is inspired by the work described in [27] and [28].

According to the proposed concept of AAO, airborne separation assurance relies on the continuous projection of the aircraft's future trajectories to identify potentially hazardous encounters. Each Autonomous Aircraft is assumed to monitor the traffic within a spherical volume of airspace centred in the aircraft with a radius equal to the ADS-B surveillance range. It is expected that the surveillance range of ADS-B could reach up to 120 nm [21]. It is assumed that the ADS-B messages received from the surrounding Autonomous Aircraft will contain intent information accurately describing their future trajectories at least until they exit AAO airspace. Based on the surveillance and intent information from the proximate traffic, each Autonomous Aircraft periodically searches for possible conflicts by comparing the proximate Autonomous Aircraft's trajectories with its own projected trajectory, which is built using its current position and its intended flight plan.

Assuming that the Autonomous Aircraft adhere to their current flight plan, the accuracy of the predictions of their future trajectories depends mainly on the following factors:

- The variability of the wind's strength and direction along the aircraft's intended trajectory.
- The aircraft's navigation and flight guidance performance.
- The quality and completeness of the position, speed and intent data included in the ADS-B messages.

- The complexity of the aircraft's intended trajectories.

The factors outlined above introduce a certain degree of uncertainty into the trajectory predictions. This hinders the performance airborne separation assurance as the accuracy of the estimation of the future distances between two aircraft depends on the uncertainty of the predicted trajectories. For simplicity, it is assumed that it is possible to define *conformance bounds* around the aircraft's nominal future positions. These conformance bounds are volumes within which an aircraft's actual position will be located with a very high probability at a certain time in the future. The size and shape of the conformance bounds depend on the degree of accuracy of the 4D FMS guidance capability and on the quality and precision of the position, speed and intent information. The conformance bounds represent the uncertainty of the trajectory prediction for a given look-ahead time.

The criterion to identify a conflict between two Autonomous Aircraft is based on the minimum predicted distance between the two aircraft, which is the distance between them at their predicted nominal closest point of approach (CPA). If this distance is smaller than an established threshold, a conflict is declared. The threshold is assumed to take into account the conformance bounds around the aircraft's nominal positions. The following three volumes are defined to justify this criterion:

- **Collision Volume**, which is a sphere of radius C centred on the aircraft. No aircraft must ever penetrate this sphere during an actual encounter, since that would be considered a near miss and could cause a collision. According to the FAA, a near miss occurs when the distance between two aircraft is smaller than 500 ft. Thus, the value of the radius C is assumed to be 500 ft. The final objective of the ASAS equipment is to provide the pilot with suitable manoeuvres to ensure that the probability of any other aircraft getting closer than 500 ft is smaller than a established Target Level of Safety.

- **Protection Volume**, which is a cylinder of radius Rp and height Hp centred on the aircraft. The successive Protection Volumes along an aircraft's projected trajectory are centred on the estimated aircraft's nominal positions. The values of R_p and S_p, which in principle may vary depending on the look-ahead time, will take into account the uncertainties in the trajectory prediction process represented by the conformance bounds introduced above. If a proximate Autonomous Aircraft's predicted future nominal trajectory is predicted to penetrate the Protection Volume, it is assumed that the probability of a violation of the Collision Volume is higher than the established Target Level of Safety and a conflict is declared. Subsequently, the conflict resolution process commences. The conflict resolution process may involve the modification of both conflicting Autonomous Aircraft's intended trajectories. The conflict resolution actions are considered to solve the conflict if the proximate aircraft's trajectory is no longer predicted to penetrate the Protected Volume. The fact that the implementation of the conflict resolution actions is predicted to result in no violations of the aircraft's Protection Volume ensures that the established Target Level of Safety is achieved.

- **Monitoring Volume**, which is a cylinder of radius R_m and height H_m centred on the aircraft. The values of R_m and H_m are slightly smaller than those of R_p and H_p respectively. The Monitoring Volume along an aircraft's trajectory is centred on the actual aircraft's future positions. If a proximate aircraft penetrates the Monitoring Volume an ASAS operational error is issued. While a small violation of the Protection Volume at the actual closest point of approach would be tolerated, a violation of the Monitoring Volume would be considered an excessively large violation of the Protection Volume and an unacceptable ASAS operation. A Target Level of Safety can also be assigned to the violation of the Monitoring Volume.

The definition of the actual sizes of the volumes described above is considered out of the scope of this work and the conflict resolution methodologies that will be proposed in this thesis are independent of these sizes. Current radar separation minima will be used to illustrate through examples the features of the proposed methodologies.

A scheme depicting the three volumes can be found in Figure 1.4. Figure 1.5 shows a diagram illustrating airborne separation assurance according to the proposed definition in a two-dimensional conflict.

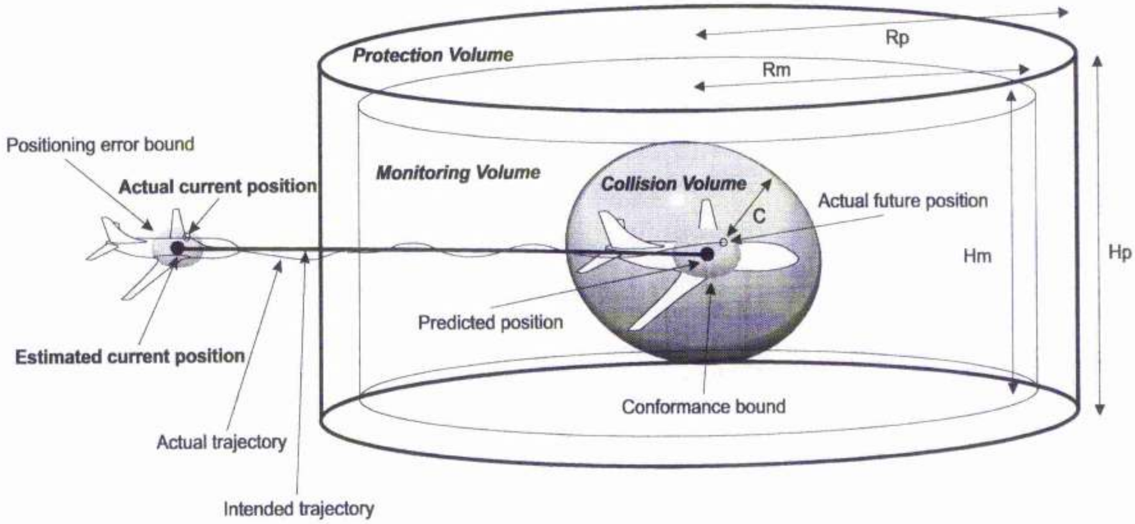


Figure 1.4: Scheme of the volumes articulating the definition of safe separation

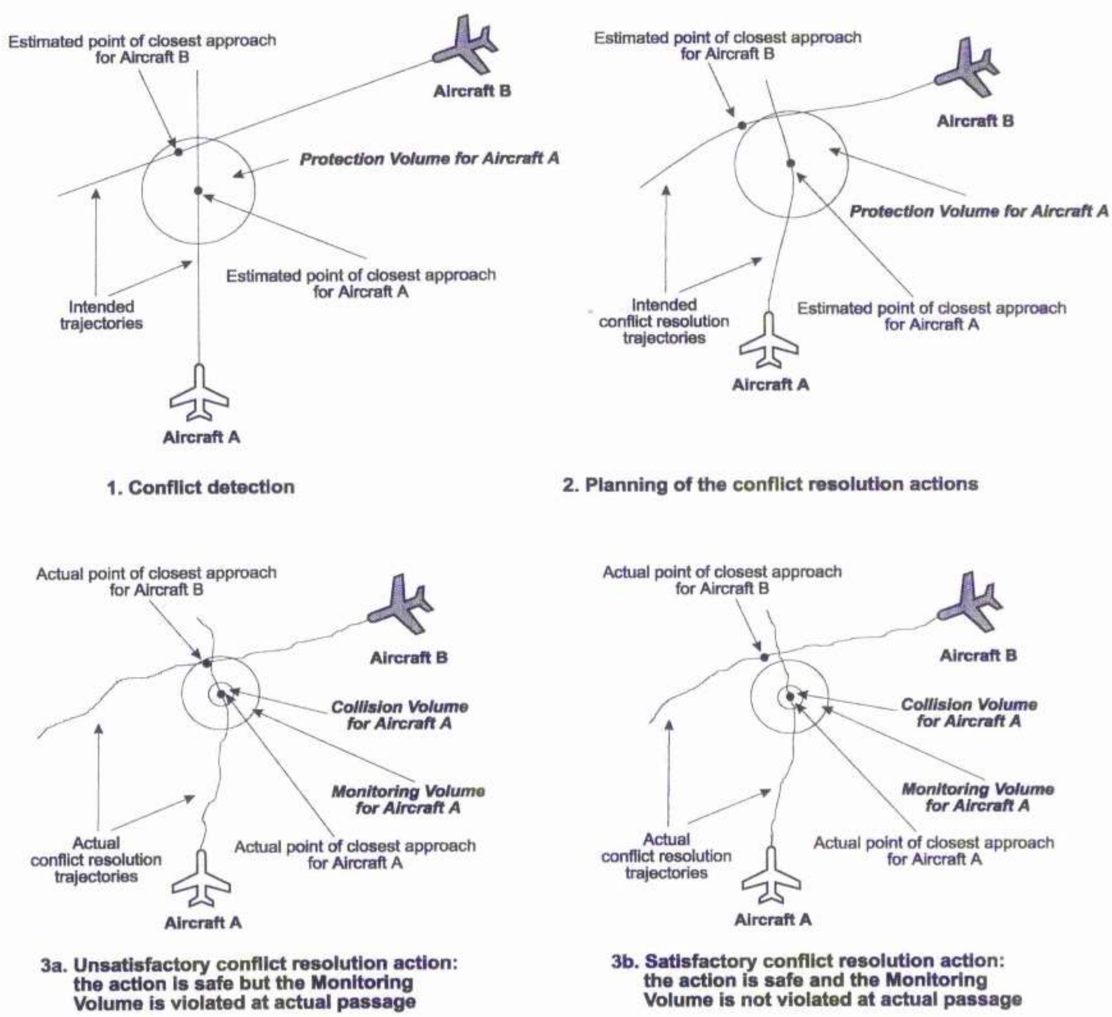


Figure 1.5: Airborne separation assurance (from the perspective of Aircraft A)

1.3.4 Cost efficient conflict resolution in AAO

To resolve conflicts, Autonomous Aircraft are compelled to deviate from their preferred routes, which causes them to incur unwanted costs. These costs are defined by the individual aircraft operator according to its preferences and can be related to the additional fuel burnt and flight time, to the delays at planned future waypoints and to the detriment of passengers' comfort, etc. Autonomous Aircraft attempt to attenuate the costs of resolving conflicts so that the benefits of flying operator-preferred routes are not entirely lost as a consequence of the performance of conflict resolution manoeuvres. Consequently, they plan their conflict resolution actions taking into account the costs of performing them. A conflict resolution action, which may involve several consecutive manoeuvres, is defined as a new intended trajectory that amends the operator's preferred flight plan.

In general, it can be expected that, as shown in [29], the closer to the CPA a conflict resolution manoeuvre is initiated, the more costly that manoeuvre has to be to safely resolve the conflict. The costs considered in [29] are related to fuel burnt and flight time. In the concept of AAO proposed here, the anticipated ADS-B range of coverage together with the assumption that the ADS-B messages include the aircraft's intent within AAO airspace, are expected to allow Autonomous Aircraft to detect conflicts early enough so that they can resolve conflicts in a cost efficient manner.

1.3.5 Co-operative approach to conflict resolution in AAO

In this thesis it is assumed that Autonomous Aircraft resolve conflicts co-operatively. As it has been said previously, co-operative conflict resolution in AAO aims to equitably distribute the resolution costs among the conflicting Autonomous Aircraft. According to this co-operative approach, Autonomous Aircraft are willing to contribute to the resolution of conflicts. They endeavour to co-ordinate their individual resolution actions with a view not only to solving conflicts safely but also to sharing the resolution costs.

1.4 Previous research relating to Autonomous Aircraft Operations

This section will provide a review of some the most relevant research on Autonomous Aircraft Operations to date. Although most of the works reviewed here do not deal explicitly with Autonomous Aircraft Operations, they are all concerned with operational concepts in which aircraft exercise responsibility for separation assurance. In general, research in this field aims to develop algorithms and operational procedures that could support airborne separation assurance with no ATC assistance.

1.4.1 The Free-route Experimental Encounter Resolution (FREER) Research and Development Programme

The Eurocontrol's FREER Research and Development Programme [30] was started in late 1995 to investigate the feasibility of the transition from a centralised ground-based air traffic control system to a distributed ground-air co-ordinated one. FREER is an ongoing programme that concentrates on investigating the delegation of separation assurance tasks from the ATC to the cockpit under the future European Air Traffic Management. In parallel, FREER also deals with the related ASAS equipment and procedures necessary to make delegation of separation assurance possible.

The main focus of the FREER programme is the Evolutionary Air-Ground Co-operative ATM Concept (EACAC), which will be briefly described below. One of the studies carried out within the FREER programme, the Full Autonomous Separation Transfer (FAST), deals specifically with airborne separation assurance in Autonomous Aircraft Operations. This study will be reviewed in detail as it proposes one of the most comprehensive conflict resolution methodologies for Autonomous Aircraft Operations to date.

1.4.1.1 Evolutionary Air-Ground Co-operative ATM Concept (EACAC)

EACAC ([31], [32]) is concerned with the investigation of the delegation of some separation assurance tasks to the pilot in near-term co-operative ASAS applications that could possibly be available by 2005 within an ATC organisation similar to the current one. EACAC proposes different sub-levels of delegation of separation assurance ranging from no delegation to limited delegation. Furthermore, EACAC proposes the concept of the *flexible use of delegation*. This concept would enable the controller to choose the appropriate sub-level of delegation for each particular traffic situation. EACAC has identified possible sub-levels of delegation for several ATC separation assurance procedures. The different sub-levels are defined by sets of tasks subject to be delegated to the pilot. The separation procedures considered by EACAC are crossing and overtaking in en-route airspace and sequencing in Terminal Management Area, which includes in-trail following and traffic merging. EACAC has also defined the operational procedures and CDTI features to support the implementation of co-operative ASAS applications for those separation procedures.

1.4.1.2 Full Autonomous Separation Transfer (FAST)

The FAST study is concerned with the full transfer of responsibility for separation assurance to the cockpit in low-density airspace. FAST specifically investigates Autonomous Aircraft Operations. Autonomous Aircraft are anticipated to operate in the future Free Flight Airspace (FFAS) regime, which is expected to be operative in some volumes of European upper airspace by 2015 and possibly elsewhere outside European airspace, such as in remote areas with no ground-based ATC coverage. According to FAST, Autonomous Aircraft's ASAS capabilities will rely on ADS-B. In addition to identification, position and velocity information, ADS-B is also required to be able to transmit aircraft intent information regarding the next trajectory change point, either automatically or by other aircraft's request.

The concept of Autonomous Aircraft Operations proposed in FAST is described in [24], [33], [23] and [34]. According to this concept, airborne separation assurance is regarded as a pilot-centred function. Thus, prototypes for a CDTI and for an ASAS pilot-machine interface are designed in addition to ASAS procedures and algorithms.

Simulations and flight trials involving pilot-in-the-loop interacting with ASAS equipment prototypes have also been carried out within FAST.

According to FAST, Autonomous Aircraft avoid conflicts predicted within a look-ahead time of 6-8 minutes. The ASAS equipment predict conflicts by projecting the trajectories of the host and proximate aircraft in 4 dimensions (4D projection): 3 space co-ordinates and time. This projection is based on the position, velocity and intent data of the host aircraft, available from the on-board systems and on the position, velocity and intent data of the proximate aircraft, received through ADS-B. The trajectory projection process also takes into account the uncertainties due to the effects of wind and the navigation inaccuracies.

A conflict is declared if the predicted distance between the host aircraft and any of the proximate ones at their closest point of approach (CPA) is smaller than the established separation minima. Conflicts are conveyed to the flight crew through aural alerts and through coloured areas displayed on the CDTI, called conflict zones. Conflict zones are to be avoided since they correspond to areas of airspace where separation minima are violated. In addition to conflict zones, no-go zones are also depicted on the CDTI. A no-go zone corresponds to a region of airspace where a loss of separation would occur if the host aircraft entered it as a result of a manoeuvre. Therefore, no-go zones represent potential conflicts. Both the conflict zones and the no-go zones are called forbidden zones. The predicted conflicts are safely resolved if all the conflict-zones are avoided without entering any no-go zones.

The conflict resolution methodology proposed in FAST is based on the assignment of a priority order to the conflicting aircraft. This priority order is established through the application of a set of rules that must be known by all the aircraft. The rules implemented in FAST are the Extended Flight Rules (EFR) [35], which assign a priority order to the conflicting aircraft based on their manoeuvrability, phase of flight, and distance to the closest point of approach. Once the priority order has been established, the aircraft with the highest priority has the right of way and maintains its intended trajectory. Then, the second aircraft in the priority order resolves its conflict with the aircraft of highest priority, ignoring its conflicts with the aircraft of lower priority. Then, the third aircraft in the priority order resolves its conflicts with the two aircraft of higher priority, ignoring its conflicts with the aircraft of lower priority. This process is repeated for all the conflicting aircraft.

Depending on the workload in the flight deck, conflicts can be resolved in three different ways:

- The pilot accepts a solution from an automatic conflict solver, which is part of the ASAS equipment. The suggested solution is incorporated into the FMS, the new intended trajectory is broadcast to the surrounding traffic through ADS-B, and the avoidance manoeuvre is implemented by the auto-pilot.

- The pilot collaborates in defining the manoeuvre computed by the automatic conflict solver by interactively adding constraints to the solution, such as the type of manoeuvre.

- The pilot manually defines a solution by introducing changes to the intended trajectory so that the modified trajectory does not intersect any forbidden zones.

Once a solution to the conflict has been found, the pilot must monitor the avoidance action to certify that the conflict is actually solved.

The FAST conflict resolution methodology has been implemented in a cockpit simulator and a series of pilot-in-the-loop simulations have been conducted to demonstrate the FAST concept for Autonomous Aircraft Operations described above [34]. Only two-aircraft encounters are considered in the experiments. Thus, the aircraft with the highest priority according to the Extended Flight Rules has the right of way and maintains its intended trajectory, while the other one solves the conflict. The pilot can resolve a conflict either automatically, by accepting one of the solutions proposed by the conflict solver, or manually, by inserting waypoints into the intended trajectory. The conflict resolution actions are restricted to lateral manoeuvres and therefore aircraft maintain their preferred altitude throughout the encounter.

The automatic conflict solver used in the simulations implements the GEARS (Generic En-route Algorithmic Resolution Service) algorithm developed by Eurocontrol [36].

The GEARS algorithm provides a set of flyable trajectories that avoid all the forbidden zones. Pilots can select one trajectory from among those in this set according to their preferred optimisation criteria. These may include minimum fuel consumption, minimum number of turns and minimum deviation from the intended trajectory.

In the simulations ACAS was available as an independent safety net for collision avoidance but ground-based ATC service was not included. The pilots participating in the experiments considered that in a real application ATC service should be available to handle emergency cases and non-nominal situations. On the other hand, they also declared that the priority-based resolution procedure induced some stressful and ambiguous situations. Such situations occurred while the pilots were monitoring a conflicting aircraft with lower priority to verify that it had actually performed a conflict resolution action. If it had not done so after some time, the pilots would try to communicate with the lower priority aircraft's pilot to clarify the situation using a voice channel. If no answer was received within a short time, pilots would initiate a conflict resolution manoeuvre even though their aircraft had the right of way. After the experiments the pilots suggested that the system could be improved if they were provided with some explicit information concerning whether a conflicting aircraft with lower priority is actually intending to manoeuvre or not. Despite these shortcomings, 60% of the pilots who took part in the simulations would be willing to accept the responsibility for separation assurance. Most of the pilots also concluded that the system would be very valuable in low-density airspace over areas with no ATC coverage.

Although the FAST conflict resolution methodology offers pilots the possibility of interacting with the ASAS equipment to select the most convenient resolution trajectory considering cost-efficiency criteria, the methodology does not provide for an even distribution of the resolution costs among all the conflicting aircraft. The application of the methodology in two-aircraft conflicts, for example, results in one of the aircraft manoeuvring and bearing all the resolution costs while the other maintains its operator-preferred route.

1.4.2 ASAS-related research at the NLR (National Aerospace Laboratory, The Netherlands)

The NLR has proposed a concept of airborne separation assurance with no ATC support based on position and velocity information broadcast through ADS-B ([37], [38], [39]). This concept introduces a future flight operating capability called Free Flight with Airborne Separation Assurance, by which the aircraft are able to select their operator-preferred routes and the flight crews hold full responsibility for separation assurance. All aircraft flying in this regime must be adequately equipped to detect and resolve the conflicts in which they may be involved. In this context, each aircraft uses the ADS-B position and intent information received from its surrounding aircraft to project their trajectories 5 minutes into the future and compare them with its own projected trajectory. If the minimum predicted distance at the closest point of approach to a proximate aircraft violates the established separation minima (5 nm horizontally and 1000 ft vertically), a conflict is declared. Considering the fact that no intent information is used, the accuracy of the conflict detection process may be degraded if a proximate aircraft initiates a manoeuvre within the next 5 minutes. To alleviate this problem, the concept of predictive ASAS (PASAS) is introduced. Predictive ASAS is a decision-support tool that advises the pilot which manoeuvres would lead to a conflict within the next 5 minutes. The interface of PASAS consists of warning colour bands superimposed on the heading, speed and vertical speeds scales of the Primary Flight Display.

The conflict resolution methodology for Free Flight with Airborne Separation Assurance proposed by NLR relies on an algorithm implemented on board all aircraft. This algorithm is based on the Modified Voltage Potential method, which was originally developed for an ATC decision-support tool [40]. The Modified Voltage Potential method relies on an analogy between conflicting aircraft and electrically charged particles. The charges are assumed to be of the same sign and, consequently, the aircraft-particles repel one another. Considering this analogy, the algorithm on board each aircraft produces an *avoidance vector* that would result in the resolution of all the predicted conflicts within the next 5 minutes if added to the current aircraft speed vector. The avoidance vector is presented to the flight crew in the form of an advised track, ground speed and vertical speed. Each conflicting aircraft calculates its avoidance vector considering that the other conflicting aircraft will not manoeuvre. However, all the conflicting aircraft intend to manoeuvre to resolve their respective conflicts. The fact that all the aircraft are equipped with the same algorithm results in avoidance vectors that contribute to steer all the aircraft away from their respective conflicts. Thus, the aircraft manoeuvre in a seemingly co-ordinated manner that helps increasing the safety of the conflict resolution methodology. For example, in a two-aircraft conflict the application of the algorithm produces avoidance vectors oriented in opposite directions. In addition to support the indirect co-ordination of their manoeuvres, these avoidance vectors allow the aircraft to resolve their conflict earlier than if only one of them applied the algorithm. Thus, despite the algorithm not taking into account cost-efficiency directly, the fact that both aircraft contribute to the resolution of their conflict allows them to share the burden of the resolution.

Prototypes of the cockpit displays and decision-support tools necessary to implement the concept of Free Flight with Airborne Separation Assurance have been incorporated into a flight simulator with the objective of conducting pilot-in-the-loop experiments. The flight crews involved in these experiments provided positive feedback about the acceptability and feasibility of the concept. However, they expressed a preference for the use of intent information and raised the issue that the performance of some resolution manoeuvres may be to the detriment of passenger comfort. They also doubted the feasibility of some of the speed control advisories considering the limited available speed regimes at cruise altitude.

While the NLR's Free Flight with Airborne Separation Assurance concept considers conflicts taking place within the next 5 minutes and detected without considering intent information, this thesis will proposed a concept of AAO in which conflict detection takes into account the intended trajectories of the conflicting aircraft. As a consequence, the look-ahead time for conflict detection will be increased significantly. According to NLR's concept, as soon as a conflict is detected, the pilot is advised to perform a certain manoeuvre that has to be initiated immediately to ensure safe separation at the CPA. On the other hand, in this thesis the longer look-ahead time for conflict detection will allow for conflict resolution actions that consist of slight modifications to the intended flight plan. These modifications will take into account cost efficiency criteria. Besides, the conflicting aircraft will knowingly attempt to co-ordinate their resolution actions with a view to sharing the resolution costs equitably.

1.4.3 Free Fight Autonomous and Co-ordinated Embarked Solver (FACES)

The FACES solver [41] is a decision-support tool for airborne separation assurance in Free Flight Airspace. The solver, which is assumed to be installed on board all the aircraft flying in this airspace regime, periodically generates a trajectory that is conflict-free with the surrounding aircraft for at least the next 5 minutes. This trajectory does not allow for a deviation from the initially intended trajectory within the first minute to allow time for the solver to compute the trajectory and for the pilot to be informed of it. The conflicts are detected using the information broadcast through ADS-B. In this case, the aircraft positions, velocities and information regarding their intended trajectories within the next 5 minutes are required. To search for possible conflicts, the solver computes the predicted minimum distance within the next five minutes between the aircraft aboard which it is installed and the other aircraft within its ADS-B range of coverage. The computation takes into account possible uncertainties in the aircraft future positions as they fly their intended trajectories. A conflict is declared when the computed minimum distance is smaller than a certain separation minimum, which is assumed to be 6 nm for the horizontal plane and 1000 ft for the vertical plane.

The conflict resolution methodology according to which the FACES solver is applied is based on the enforcement of a resolution order among all the conflicting aircraft. Once a resolution order has been established, the first one applies the solver to generate its new intended trajectory without considering the other conflicting aircraft. Hence, the first aircraft does not need to alter its initially intended trajectory. Then the second aircraft applies the solver to generate a new intended trajectory that is conflict-free with the intended trajectory of the first one. Subsequently, the next aircraft applies the solver to generate its new intended trajectory so that is conflict-free with the intended trajectories of the two previous aircraft, and so on. While in FAST the aircraft are ordered according to their respective priorities, which are given by applying the Extended Flight Rules, in FACES the order is defined according to a *token allocation* *strategy*. This strategy ensures that the FACES conflict resolution methodology, unlike the FAST methodology, is successfully applicable in situations where one aircraft is simultaneously in conflict with two other aircraft that are outside the ADS-B range of coverage of one another.

To generate new intended trajectories, the FACES solver uses the classical A^* search algorithm, which finds a minimal cost path joining the start node and the end node of a graph [42]. The cost function considered takes into account the length of the resulting trajectory as well as the efficiency of the manoeuvres that the aircraft have to perform. Although it takes into account the conflict resolution costs, the FACES conflict resolution methodology does not attempt to distribute these costs among the conflicting aircraft.

The FACES conflict resolution methodology is applicable in scenarios with high air traffic density. Simulations have also demonstrated that the NLR's Free Flight with Airborne Separation Assurance concept could be applied in such scenarios [43]. However, the reactive nature of the NLR's conflict resolution methodology produces greater disruptions in the traffic flows than the FACES methodology, which takes into account intent information and manoeuvre efficiency.

The implications of the FACES conflict resolution methodology for the flight crew have not been analysed in depth. The methodology assumes that a new intended trajectory is produced by the solver and accepted by the crew every minute. The fact that the aircraft intent is liable to change every minute could result in an excessive strain for the flight crew. The FACES conflict resolution methodology does not consider a model of the flight crew response latency and does not provide for non-nominal situations where the flight crew takes longer to accept the trajectory or considers that the trajectory produced by the solver is not appropriate.

1.4.4 Other research relating to airborne separation assurance with no ATC assistance

1.4.4.1 Optimisation of manoeuvre co-ordination rules

The work presented in [44] and [45] proposes a method based on Genetic Algorithms [46] to obtain an optimal combination of rules to support a rule-based conflict resolution methodology for airborne separation assurance with no ATC assistance. A rule-based conflict resolution methodology is one that relies on the enforcement of a resolution order according to a given set of priority rules. The FAST conflict resolution methodology can be classified as rule-based, as it is based on the application of the Extended Flight Rules to establish a priority order among the conflicting aircraft. Given the set of rules that can be applied in a certain conflict scenario and considering the resolution manoeuvres available to the conflicting aircraft once a resolution order has been established, the application of the proposed method results in a combination of rules that is optimal according to certain safety, efficiency and rule complexity criteria.

1.4.4.2 Hybrid control-based approach to airborne separation assurance with no ATC assistance

The work described in [47], [48], [49], [50] and [51] focuses on the development of a method based on hybrid control to verify the safety of conflict resolution manoeuvres for airborne separation assurance with no ATC assistance. Additionally, the method facilitates the synthesis of control schemes that could enable the aircraft to implement the resolution manoeuvres that are proven to be safe. According to the proposed method, conflict resolution manoeuvres are modelled adopting a hybrid system perspective. The performance of a resolution manoeuvre requires the aircraft to switch between different pre-defined *flight modes* such as constant airspeed, constant heading and constant bank angle. Each flight mode is characterised by specific continuous aircraft dynamics.

The proposed hybrid control method is considered as a possible foundation for conflict resolution methodologies that could support airborne separation assurance with no ATC assistance. In an operational concept where the conflicting aircraft do not attempt to collaborate with one another in resolving their conflicts the method could be used to enable a conflicting aircraft to design a resolution manoeuvre that is safe regardless of the actions of others. In this case, the conflict resolution methodology in place is referred to as *non-co-operative.* In an operational concept where the conflicting aircraft collaborate with one another in the resolution of their conflicts, a *co-operative* conflict resolution methodology could be devised based on the proposed hybrid control approach. According to such a conflict resolution methodology, the conflicting aircraft would co-ordinate their actions by implementing a combination of pre-defined manoeuvres that is proven to be safe in the specific conflicting configuration. Each aircraft would need to know which type of resolution manoeuvre to perform in that configuration as well as which values of the parameters defining the manoeuvre are guaranteed to be safe. For examples, those parameters could be the switching times between flight modes and the value of the angular and lineal velocity during each mode.

1.4.4.3 Experimental work relating to airborne separation assurance with no ATC assistance

The objective of the experimental work described in [52] was to identify the human factor issues related to airborne separation assurance with no ATC assistance. To do so, a pilot-in-the loop simulation was conducted to analyse the performance of flight crews in traffic situations where they have to maintain safe separation from their surrounding aircraft. During the simulations, no ATC support was available and inter-crew communications were not allowed. The cockpit simulator was equipped with a CDTI that included a proximal aircraft warning system but no other decision support tools for conflict detection and resolution were provided. The crews had to become aware of possible conflicts and elaborate and implement a strategy to resolve them. Thus, there was no pre-defined conflict resolution methodology in place. In view of the performance of the flight crews in the simulations, it was concluded that good situation awareness was not sufficient to ensure safe separation. It was also suggested that airborne alerting systems should be designed taking into account pilot's knowledge and operating procedures.

For the pilot-in-the loop simulations described in [53], the airborne alerting logic described in [54] was implemented together with a CDTI in a cockpit simulator. This alerting logic defines four different stages of alert depending on the probability of conflict and on the avoidance manoeuvres available to the flight crew. The probability of conflict is estimated from position and velocity reports broadcast through ADS-B using Monte-Carlo simulations, in which navigation errors and unexpected manoeuvres are treated as random uncertainties in the aircraft projected trajectories. The traffic scenarios for the pilot-in-the-loop simulations were set up so that the participating crew's aircraft would conflict with another aircraft if it did not alter its planned route. The flight crews taking part in the simulations were instructed to determine which aircraft should manoeuvre to resolve a predicted conflict by applying the standard right-of-way rules used in VFR (Visual Flight Rules) operations [12]. Once the priority had been established, the flight crew of the aircraft that gave way had to decide which resolution manoeuvre to perform. Unlike the experiments described in [52], in this case the flight crews were allowed to communicate with one another and even with ATC if they considered it necessary. A confederate pilot and a confederate controller assisted in the communications and negotiations with the flight crews during the pilot-in-the-loop simulations.

The simulations showed that most flight crews contacted the flight crew of the conflicting aircraft. On the other hand, very few flight crews contacted ATC. Considering the findings regarding inter-crew communications, it was concluded that the use of the voice channel for the co-ordination of resolution manoeuvres could lead to frequency congestion in a practical implementation of the airborne separation assurance concept proposed. In the simulations it was also observed that many flight crews decided to perform a resolution manoeuvre even in situations when they had the right of way, as had also occurred during the FAST pilot-in-the loop simulations. Uncertainty about the actions of the subordinate conflicting aircraft may lead the flight crew of the aircraft with the right of way to manoeuvre with a view to ensuring safety.

1.4.4.4 Research relating to conflict resolution algorithms for airborne separation assurance with no ATC assistance

In the work described in [55] and [56] the artificial potential fields approach is applied in the development of a conflict resolution algorithm for self-separating aircraft. Each conflicting aircraft determines a lateral resolution action from an artificial force derived from a potential function. This potential function is defined based on the predicted distance between the aircraft at their CPA. Assuming that all the conflicting aircraft apply the algorithm, the resulting manoeuvres allow for the co-ordinated resolution of the predicted conflicts without the need for inter-aircraft communication. The algorithm does not consider the aircraft's intended trajectories and the distance between the aircraft at their CPA is calculated by extrapolating the current positions based on the current speeds. Thus, conflicts cannot be predicted too far in advance because of the likelihood of the aircraft changing their speed or course before the CPA, which in turn increases the likelihood of a defective prediction. The algorithm does not consider explicitly the cost efficiency of the resolution manoeuvres.

The work described in [57] concentrates on the development of a method to estimate the probability of conflict between aircraft. This method is based on a model of the aircraft's actual future trajectories that consists of the aircraft's nominal future trajectories perturbed by a Brownian motion. The Brownian motion represents random uncertainties such as those due to the effects of the wind and navigation inaccuracies. No intent information is considered in the synthesis of the aircraft's nominal future trajectories, which are the result of projecting the current positions and velocities into the future. This method to estimate the probability of conflict has been specifically designed for two-dimensional encounters involving two or more aircraft flying straight lines at constant speeds. In this context, a conflict is deemed to occur when the distance between two aircraft is smaller than 5 nm. The approximate probability of a conflict occurring is computed using results from the theory of Brownian motion regarding boundary crossing.

An algorithm that enables aircraft to generate conflict-free trajectories is proposed, based on the method for estimating the probability of conflict. The algorithm periodically calculates the probability of conflict with the proximate aircraft. Based on these calculations, it produces a sequence of heading changes that steers the aircraft towards zones with lower probability of conflict and eventually guides it back to its initially intended route. The trajectories that result from simulating the application of the algorithm have been shown to be conflict-free in several examples, but the algorithm has not been formally proven to be safe. In the simulations, the aircraft are assumed to automatically implement the heading changes output by the algorithm. Hence, the role of the flight crew in the conflict resolution process is not considered. Additionally, the trajectories obtained in the examples presented involve continuous heading changes and do not take into account efficiency criteria.

The research presented in [58] considers the problem of synthesising conflict-free trajectories in a free routing environment as a multi-participant optimal control problem. Aircraft involved in a potential conflict are thought to attempt to optimise their performance while maintaining separation from their proximate aircraft. Each aircraft defines a performance index that reflects the penalty of deviating from its operator-preferred route to resolve conflicts. Two different approaches are proposed to solve this optimal control problem. In the first approach, a piece-wise linear trajectory parameterisation is used to convert the problem into a parameter optimisation problem. Accordingly, aircraft intended trajectories are modelled by a set of four-dimensional waypoints, which consist of three position co-ordinates and time. The waypoints are assumed to be connected by straight lines. Considering this trajectory parameterisation, both a single-objective and a multiple-objective optimisation problem are formulated.

In the single-objective formulation, the conflicting aircraft attempt to synthesise new intended trajectories that resolve their conflicts while minimising a global performance index that is given by the sum of the integral deviations of each conflicting aircraft from its initially intended trajectory. An algorithm based on the Sequential Quadratic Programming (SQP) method, which is suitable for optimising continuous non-linear objective functions, is used to adjust the parameters defining the waypoints so that the resulting trajectories minimise the performance index while meeting the aircraft's performance constraints and, most importantly, remaining conflict-free.

In the multiple-objective formulation, each aircraft attempts to minimise its own individual performance index that reflects the cost-efficiency criteria of its operator. To solve the resulting problem, the goal attainment method is applied. The individual objectives and the constraints are considered as goals to be satisfied. The degree to which these goals are to be met is adjusted using a vector of weighting factors, which expresses a measure of the relative trade-offs among the objectives. Given a vector of weighting factors, the goal attainment method is posed as a non-linear single-objective optimisation problem, which is also solved using an algorithm based on the SQP method.

The single-objective and the multiple-objective formulations described above have been shown to result in conflict-free resolution trajectories in different conflict scenarios involving up to 6 aircraft. However, some of the resolution trajectories obtained would require the aircraft to perform successive climbs and descends, which might be undesirable in practice. Although the two formulations have been successfully applied in the scenarios considered, none of them are guaranteed to result in a solution.

The second approach proposed in [58] to solve the multi-participant optimal control problem, consists of developing closed-loop guidance laws indicating the perturbations the aircraft have to introduce on their nominally intended trajectories to resolve their predicted conflicts. An algorithm to develop such guidance laws for conflicts involving two-aircraft has been proposed. The algorithm, which is based on the neighbouring extremal method for optimal control problems, results in a closed-loop guidance law that approximately minimises a global performance index relating to the deviations of the two aircraft from their nominal trajectories caused by the perturbations. The performance of the algorithm is illustrated in a two-aircraft conflict scenario. The implementation of the guidance laws obtained result in both aircraft modifying their trajectories to resolve the conflict.

Assuming that an aircraft has access to the intended trajectories and cost-efficiency criteria of the proximate aircraft, that aircraft could, when in conflict with others, use any of the three algorithms proposed in [58] to alter the intended trajectories of each of the conflicting aircraft so that their conflicts are resolved. The implementation of the new trajectories would result in each of the conflicting aircraft contributing to the conflict resolution process and bearing part of the costs involved in it. A hypothetical conflict resolution methodology based on any of the three algorithms above would have to include the necessary means to ensure that all the conflicting aircraft are informed of their new trajectories and agree to implement them. Thus, a high-performance data-link would have to be in place to enable the aircraft to exchange trajectory-related and other information. Additionally, the necessary decision-support tools, communication protocols and operational procedures would have to be designed to enable flight crews to become aware of the conflict resolution process and remain in control of it.

1.5 Objectives and contributions of the research described in this thesis

The main objective of the work described in this thesis is to investigate the potential application of Distributed Artificial Intelligence concepts and techniques to support co-operative conflict resolution in AAO. As stated previously, the term 'co-operative' in this context is used to describe a form of resolving conflicts in which Autonomous Aircraft collaborate with one another to co-ordinate their resolution actions and to share the costs involved in the conflict resolution.

A new approach based on multi-agent systems has been developed to study conflict resolution in AAO. In this new approach, Autonomous Aircraft are modelled as agents and conflict resolution is regarded as co-operative activity in the framework of a multi-agent system. Co-operation in a multi-agent system refers to the agents interacting with one another to co-ordinate their actions so that those actions do not conflict with one another and that satisfactory global performance is achieved. The means necessary for the agents in a multi-agent system to co-operate with one another are provided by a *co-operation mechanism*, which could involve, among other elements, planning algorithms, behaviour rules and communication protocols. Based on a review of relevant research literature, two main types of co-operation mechanisms in multi-agent systems have been identified: *behaviouristic* and *reflective.* The former type refers to co-operation mechanisms that result in agents seemingly acting in a co-operative manner according to the judgement of an external observer, regardless of whether or not the agents are knowingly co-operating. The latter type refers to co-operation mechanisms that enable agents to engage knowingly in what they consider as co-operative activity, regardless of whether or not they are seen as co-operating by an external observer.

To illustrate the potential of the proposed DAI-based approach to conflict resolution in AAO, two co-operation mechanisms for a multi-agent system of conflicting Autonomous Aircraft have been devised. Both co-operation mechanisms are inspired by research on multi-agent systems and are proposed as co-operative conflict resolution methodologies for AAO. One of the two mechanisms is of the behaviouristic type and has been designed specifically for an operational environment where the only means for Autonomous Aircraft to exchange information with one another is through ADS-B. The other mechanism is of the reflective type and has been designed specifically for an operational environment where Autonomous Aircraft are able, not only to broadcast information through ADS-B, but also to exchange information with one another on a one-to-one basis through a point-to-point data-link. In both operational environments, the corresponding co-operation mechanism result in a conflict resolution methodology that enables Autonomous Aircraft to co-ordinate their conflict resolution trajectories so that they can resolve their conflicts safely while sharing the costs of flying those trajectories. Each of the mechanisms include the algorithms, protocols and procedures necessary to enable flight crews, aided by the appropriate decision-support-tools, to apply successfully the conflict resolution methodology defined by the mechanism.

1.6 Outline of the remainder of this thesis

Chapter 2 introduces co-operation in multi-agent systems and proposes a multi-agent-based model of the operational concept adopted for Autonomous Aircraft Operations. This chapter also introduces the two co-operation mechanisms for conflict resolution in AAO that have been developed based on this model.

The behaviouristic co-operation mechanism is explained in detail in chapter 3, which also describes a practical implementation of this mechanism for two dimensional conflict scenarios involving up to three aircraft. In these types of conflict scenarios, the conflict resolution methodology that results from the implementation of the mechanism is shown to enable conflicting Autonomous Aircraft to co-ordinate their resolution trajectories so that their conflicts are resolved and so that the resolution costs are distributed among them. The mechanism, inspired by the Recursive Modelling Method for co-ordinating the actions of the agents in a multi-agent system, results in a conflict resolution methodology by which, one after another, the conflicting aircraft apply the same trajectory-planning algorithm to synthesise their resolution trajectory. As soon as the flight crew accepts the implementation of the resolution trajectory, the new intent is broadcast through ADS-B. The order in which the aircraft elaborate their resolution trajectory is random and depends on the acceptance times of the flight crews. The trajectory-planning algorithm is based on Game Theory, particularly on Stackelberg games, and produces a resolution trajectory that is conflict-free with those already broadcast by other conflicting aircraft and is simultaneously expected to facilitate the process of synthesising a resolution trajectory for the conflicting aircraft that remain to do so.

Chapter 4 describes the proposed reflective co-operation mechanism, which is inspired by research on team-based co-operation in multi-agent systems. The mechanism has been designed for an operational environment in which Autonomous Aircraft can communicate with one another through a point-to-point data-link. This communication capability enables conflicting aircraft-agents to form teams and act together towards the resolution of their conflicts. The members of a team agree on a joint plan to resolve their conflicts and commit to the implementation of their respective actions in that plan. One of the team members, the *team organiser*, applies a centralised planning algorithm to generate the conflict resolution trajectories of all the team members. The proposed reflective co-operation mechanism includes the procedures and protocols necessary to form teams as well as a planning algorithm to elaborate the conflict resolution plans. The team formation process that results from the implementation of the mechanism is described in detail. A planning algorithm based on multi-objective combinatorial optimisation techniques is proposed. The performance of the mechanism with this algorithm is analysed for the same type of conflict scenarios as in Chapter 3. The analysis shows that the proposed mechanism enables Autonomous Aircraft to co-ordinate their resolution actions so that they share the resolution costs. The performance of the reflective co-operation mechanism is compared to that of the behaviouristic one in the scenarios studied.

Chapter 5 discusses some conclusions regarding the work described in this thesis and makes recommendations for further research.

The thesis ends with two appendices. Appendix A outlines the Airborne Collision Avoidance System (ACAS). Appendix B overviews the different data-link technologies that are being considered to support the implementation of ADS-B.

Chapter 2

Multi-agent systems and their application in Autonomous Aircraft Operations

2.1 Introduction

This chapter proposes a new approach for the development of co-operative conflict resolution methodologies for Autonomous Aircraft Operations. The approach is based on modelling conflict resolution among Autonomous Aircraft in terms of the interactions between the agents in a multi-agent system. The contents of this chapter are as follows. Firstly, the disciplines of Artificial Intelligence and Distributed Artificial Intelligence are introduced, focusing on the concept of agent and on the study of multi-agent systems. Secondly, the two main types of co-operation mechanisms in multi-agent systems, behaviouristic and reflective, are explained using examples from relevant research literature. Subsequently, a model of Autonomous Aircraft Operations based on multi-agent systems is presented. In this model, conflicting Autonomous Aircraft are considered as agents that co-operate with one another in the framework of a multi-agent system to resolve their conflicts. A co-operation mechanism provides the means by which aircraft-agents co-operate, thereby defining the conflict resolution methodology in place. To conclude this chapter, two examples of co-operation mechanisms for the multi-agent system considered, are introduced. Each of these mechanisms has been specifically designed for a different operational environment where the proposed operational concept for AAO could be implemented. In the next chapters, those two co-operation mechanisms will be described in detail.

2.1.1 Artificial Intelligence (AI)

The field of Artificial Intelligence (AI) endeavours to understand and build intelligent entities [59]. Traditionally, one of the main motivations for AI research has been to achieve a better understanding of the human mind. In fact, AI still retains an element of speculation about human intelligence and has inherited theories, ideas and techniques from disciplines such as philosophy, psychology and linguistics. The widespread use of computing has transformed AI into a multi-disciplinary field encompassing not only theoretical but also experimental research. Currently, AI comprises of a huge variety of sub-fields, ranging from general-purpose areas, such as perception and logical reasoning, to specific applications such as disease diagnosis, mobile robotics and chess-playing machines. Furthermore, scientists from other fields often move into AI and use its tools and vocabulary to analyse, systematise and automate complex behaviours.

Traditionally, four different approaches to Artificial Intelligence have been followed since the term was coined in 1956. They correspond to four different interpretations of the concept of an intelligent entity, which is the subject matter of AI. The four approaches can be succinctly described as follows [59]:

- **Intelligent entities as systems that think like humans:** The design of systems that think like humans inevitably involves understanding how humans think. Becoming familiar with the human mind can be accomplished by introspection. This involves analysing our thoughts and the way in which mental processes occur, or through psychological experiments. The field of cognitive science strives to construct working models of the human mind. Once a satisfactory theory of the mind is developed, the aim of AI would be to create computer-based systems that mimic the human reasoning process and produce identical results.

- **Intelligent entities as systems that think rationally:** Researchers working within this approach to AI endeavour to build computer programs capable of finding solutions to real problems expressed in logical notation by applying the laws of logic. The developments in the field of formal logic in the late nineteenth and early twentieth centuries fostered this approach to AI. In as much as the rules of formal logic can be regarded as a model of human rational thought, computer programs with the ability to solve problems through the application of those rules are considered as intelligent entities.

- **Intelligent entities as systems that act like humans:** Within this approach to AI, intelligence is defined as the ability to achieve human-level performance in cognitive tasks. Researchers following this approach assume that an intelligent entity should be able to successfully interact with humans at their same level. This interaction would involve understanding their language, drawing conclusions, providing understandable answers to their questions, and learning to adapt to new circumstances.

- **Intelligent entities as systems that act rationally:** Researchers following this approach to AI consider intelligent entities as systems capable of perceiving the environment around them and acting rationally in that environment. These systems are called rational or intelligent agents. Thus, in this approach AI is viewed as the study and construction of rational agents. An agent is considered to act rationally if it achieves a certain degree of success in its environment. The degree of success is usually evaluated through an established performance measure. Within this approach, agents are often viewed as intentional systems [60] [61], which are entities whose behaviour is described by ascribing them human-like mental attributes such as beliefs, desires and intentions.

Different definitions of Artificial Intelligence are given in Figure 2.1.

Concept of intelligent entity	Definitions of Artificial Intelligence
Systems that think like humans	▪ "The exciting new effort to make computers think. [...] *machines with minds*, in the full and literal sense." [59] • "[The automation of] activities that we associate with human thinking activities such as decision-making, problem solving, learning [...]." [59]
Systems that think rationally	• "The study of mental faculties through use of computational models." [59] • "The study of the computations that make it possible to perceive, reason and act." [59]
Systems that act like humans	• "The art of creating machines that perform functions that require intelligence when performed by people." [59] • "The study of how to make computers do things at which, at the moment, people are better." [59]
Systems that act rationally	• "A field of study that seeks to explain and emulate intelligent behaviour in terms of computational processes." [59] • "The branch of computer science that is concerned with the automation of intelligent behaviour." [59]

Figure 2.1: Different definitions of Artificial Intelligence according to the four different interpretations of the concept of intelligent entity

The agent approach to AI is adopted in this thesis to study Autonomous Aircraft Operations. The concept of agent is explained in detail below.

2.2 The concept of agent

In principle, an agent is anything that can be viewed as perceiving its environment and acting *rationally* upon it [59]. An agent's actions are rational if they cause it to be successful in its environment. The degree of success of the agent is generally evaluated using a performance measure established by an external observer. A human being can be considered as an agent that interacts with its environment through organs and other body parts. A robot can be seen as an artificial agent placed in the physical world that uses cameras and other sensors to perceive its surrounding environment and actuators to act upon it. A computer program can also be viewed as an agent that interacts with its software environment by exchanging messages and commands.

Agents, which are also referred to as *intelligent agents* or *rational agents*, are expected to react successfully to changing environment conditions without external intervention. They are usually equipped with built-in knowledge about their environment, which assists them in the process of making the adequate decisions about which actions to perform. Agents may be capable of interacting with other agents, mainly in the form of communication via an agent communication language [62], and may also be able to interact with humans through the appropriate interface [63].

The main objective of AI, with regards to agents, is to design and build *agent programs* [59]. The agent program implements the conceptual model of an agent, including its properties and specifications. It encompasses the internal data structures and decision-making procedures that determine the behaviour of the agent in its environment and enable it to respond to its perceptions. In general, the agent program runs on a computer system, which is called the *agent architecture* [59]. The architecture makes the agent's perceptions available and understandable to the agent program, runs the program and supplies the program's outputs to the agent's interface with the environment.

Agents can be classified into two broad types, each of which corresponds to a different approach to agency: *knowledge-based agents* and *reactive agents*. The main characteristics of the two types are described below.

2.2.1 Knowledge-based agents

Knowledge-based agents are agents that explicitly know about their environment and are somehow capable of reasoning which courses of action to take [59]. They incorporate built-in knowledge about how their environment works and how their actions affect it. They keep track of the evolution of that environment through internal representations of its changing states. Knowledge-based agents are able to adapt to changes in their environment because of their capability to increase and update their knowledge about it. Since they are capable of anticipating the effects of their actions upon their environment, knowledge-based agents can commit to achieve goals, through the implementation of their chosen actions. A diagram representing the agent program of a basic knowledge-based agent is depicted in Figure 2.2.

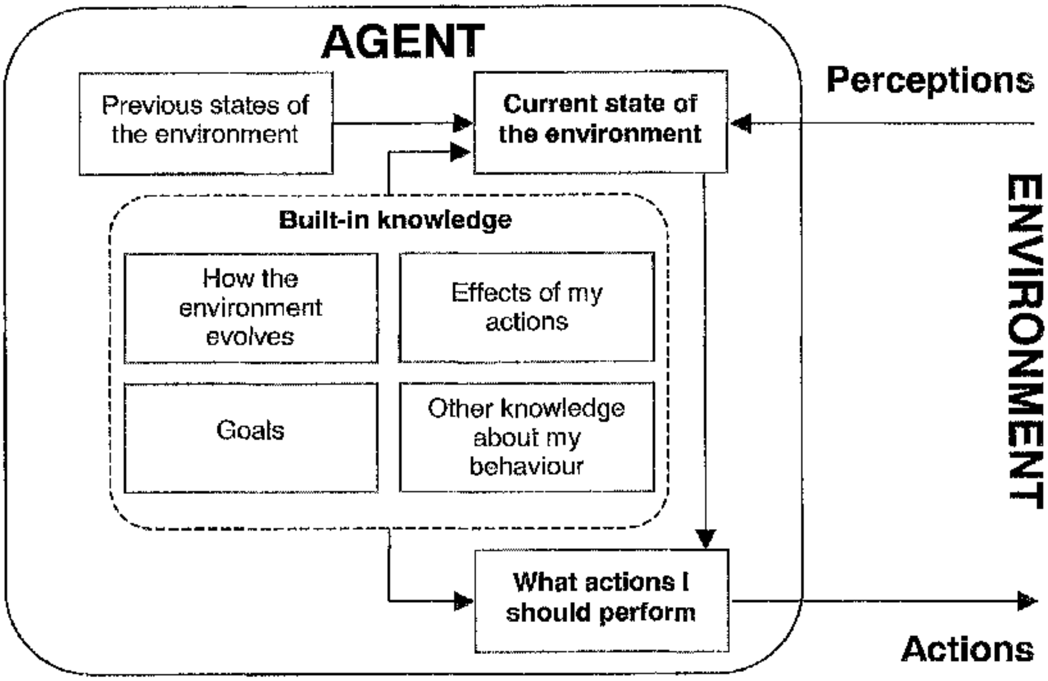


Figure 2.2: Agent program of a basic knowledge-based agent.

The set of internal representations "known" by a knowledge-based agent is referred to as its *knowledge base.* The agent's knowledge base contains facts about itself, its environment, and its possible interactions with its environment. It comprises of the agent's knowledge about the current, past and future states of the environment as well as knowledge relating to the way in which the environment evolves. It also includes the agent's possible actions in its environment as well as the agent's goals in that environment. The knowledge base can be encoded using a *knowledge representation language,* which expresses knowledge in a formal and structured form that can be manipulated by the agent program. The knowledge-based agents that are capable of expressing their knowledge in a knowledge representation language and reasoning about that knowledge using that language are called *deliberative agents.*

The basic meaningful unit in a knowledge representation language is the *sentence*. A sentence encodes a piece of the agent's knowledge. As in natural languages such as English, knowledge representation languages consist of *syntax* and *semantics*. The syntax is the set of rules that establish how sentences are to be formulated. It refers to the physical construction of proper sentences by using a set of symbols manageable by the agent program. The semantics refers to the facts in the agent's environment to which sentences refer. It establishes a connection between sentences and facts. Thus, the semantics provide sentences with meaning and enable the agent to believe the facts expressed by them.

Deliberative agents construct a symbolic model of their environment by representing it as a set of sentences expressed in a knowledge representation language. They are able to reason explicitly about these sentences by applying an *inference procedure*, which generates new sentences from known sentences. The new sentences correspond to facts that logically follow on from the facts expressed by the known sentences. The relationship between the known sentences and the new ones constructed by applying the inference procedure is called *entailment*. Entailment mirrors the property of facts that follow on from other facts by ensuring that the sentences generated by the inference procedure are true, given that the known sentences are true. The connection between facts following on from other facts and sentences entailing other sentences is depicted in Figure 2.3.

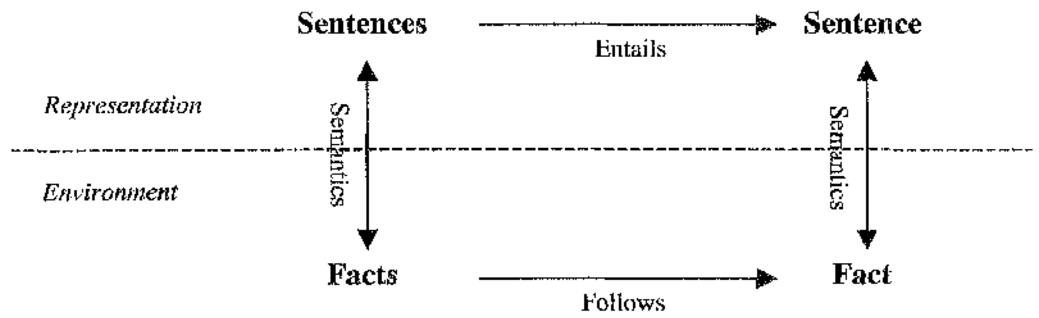


Figure 2.3: Logical inference in deliberative agents.

The set of sound inference procedures that can be applied to sentences expressed in a given knowledge representation language is called the *proof theory*. Once the proof theory has been defined, that language is called a *logic* ([59], [64]). Deliberative agents increase their knowledge about their environment by applying the proof theory to sentences contained in their knowledge base.

Non-deliberative knowledge-based agents do not use an internal logic to represent knowledge and perform explicit reasoning. However, their behaviour is explained according to the knowledge-based approach to agency. They are considered capable of internally representing knowledge and making reasonable decisions, regardless of how their agent program actually implements these capabilities. They are seen as keeping track of their environment and searching for courses of action to achieve goals. The knowledge base of a non-deliberative knowledge-based agent is a conceptual abstraction that models internal functions implemented in the agent program. The agent program may apply diverse computational techniques to generate the behaviour specified by the agent's designer. Logics can still be used as external languages to formally model, specify and design behavioural properties of non-deliberative knowledge-based agents [61].

An example of a knowledge-based agent is a *planning agent* ([59], [64], [65]). A planning agent devises sequences of actions that achieve its goals and executes them. Planning agents can be either deliberative or non-deliberative. A deliberative planning agent's agent program contains explicit logical representations of the possible states of the environment, the agent's goal states, its available actions and their effects upon the environment. It also incorporates logical inference procedures that enable the agent to search for a sequence of actions that leads to the attainment of its goals. On the other hand, a non-deliberative planning agent searches for sequences of actions that achieve its goals without using an internal logic. The agent's agent program encodes a representation of the agent's knowledge about its environment, goals, possible actions and their effects upon the environment. It also incorporates computational algorithms capable of devising a plan to realise those goals. However, the agent program does not implement an internal logic and therefore it supports neither logical knowledge representation nor explicit reasoning.

Knowledge-based agents are often viewed as *intentional systems* [60]. An intentional system is an entity whose behaviour is described by ascribing it anthropomorphic mental attributes such as beliefs, desires and intentions. Thus, knowledge-based systems can be attributed human-like mental states and attitudes to explain their behaviour. This approach to the study and design of agents is described as taking the *intentional stance* ([60], [64], [62], [61]). In principle, both deliberative and non-deliberative knowledge-based agents can be modelled taking the intentional stance. According to the intentional stance, an agent's behaviour is not only determined by its knowledge about its environment but also by its internal mental state. Thus, the agent's knowledge base incorporates representations of their mental attributes. Typical mental attributes used in the design of intentional systems are *belief*, *desire*, *intention* and *commitment*. Regardless of how these attributes are actually implemented in the agent's agent program, they represent a powerful tool to model high-level cognitive specifications for knowledge-based agents.

2.2.2 Reactive agents

Reactive agents do not incorporate a symbolic model of their environment, nor do they carry out internal reasoning processes. Instead, they merely respond to external perceptions according to basic pre-defined behaviours ([64], [62], [66]). However, reactive agents may be able to display complex rational behaviour, which would emerge as a combination of simple pre-defined responses. In this case, rationality is viewed as an attribute of the agent's behaviour identified by an external observer. Reactive agents may include simple internal representations about their environment required to select their actions, but do not perform any kind of internal reasoning based on those representations. They are behaviour-based entities and therefore simply act. Hence, they neither commit to the attainment of goals nor plan ahead.

A reactive agent's agent program usually consists of a set of *behavioural modules*. Each behavioural module assesses the agent's perceptions independently and elaborates actions that respond to those perceptions according to a pre-established simple behaviour. Behavioural modules are organised into a prioritised hierarchy in which higher level behaviours can modify and override lower level behaviours [59]. The implementation of these agent programs result in simple agents being capable of reacting quickly to changes in their environment. This is because they do not have to carry out complex reasoning, but merely simple computations. A diagram representing a model of the agent program of a basic reactive agent is shown in Figure 2.4.

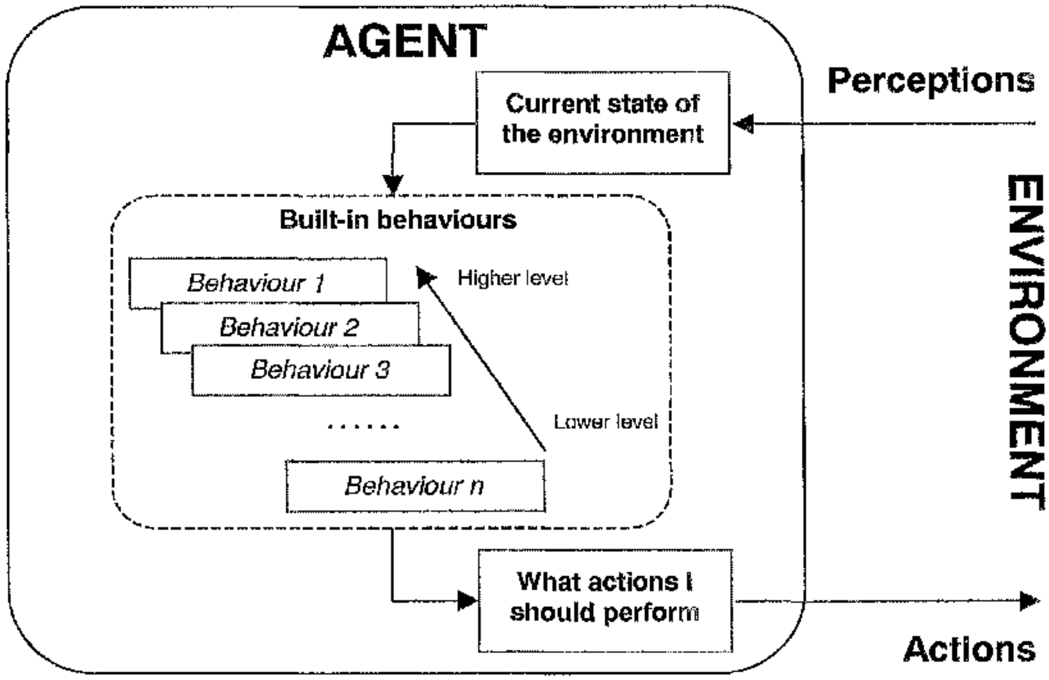


Figure 2.4: Agent program of a basic reactive agent.

As an example of a reactive agent, consider a mobile robot whose overall behaviour is the result of the combination of a set of pre-defined basic behaviours, organised into a hierarchical layered structure. Suppose the robot is in a room and one of the robot's pre-defined behaviours is simply to roam around the room. A higher level basic behaviour can be to identify an exit and leave the room through it, which implicitly encodes an agent's goal established by the designer. When the robot comes across an exit, this behaviour would prevail over roaming and the robot would leave the room through that exit. An additional basic behaviour, such as obstacle avoidance, may be implemented at the highest level to override any other basic behaviour if an imminent collision with an object in the room is anticipated. When the robot finds an exit and successfully leaves the room, its overall behaviour can be viewed as rational by an external observer.

The reactive approach to agency appears to directly contradict the knowledge-based approach. However, work has been carried out to build agents that combined elements from both approaches [64]. Such agents are called *hybrid agents* and their agent programs take advantage of the reactive approach to quickly and effectively respond to changes in their environment while supporting some kind of reasoning based on a symbolic model of the environment.

2.3 Distributed Artificial Intelligence (DAI) and the study of multi-agent systems

Agents are generally deployed in an environment containing other agents, with which they are expected to interact. The discipline concerned with the study of these interactions is Distributed Artificial Intelligence (DAI). One of the approaches adopted by Distributed Artificial Intelligence to analyse the interactions among agents consists of viewing the agents as constituting a multi-agent system. A multi-agent system is a group of two or more agents that *co-operate* with one another ([64], [60], [66], [67]). Generally speaking, co-operation can be defined as to acting with others towards a common purpose or benefit [68]. In the context of multi-agent systems, the concept of co-operation can be interpreted in different ways. This interpretation depends upon the nature and structure of the agents in the system, their capabilities and skills in their environment, their individual goals and the goals of the system as a whole and the available means of communication among agents. Co-operation in a multi-agent system generally involves the agents interacting with one another to co-ordinate their actions so that those actions do not conflict with others in the group and so that satisfactory global performance is achieved [69]. Additionally, co-operation is expected to result in some kind of global performance gain not achieved by the agents acting independently.

The type of co-operation implemented in a multi-agent system is determined by an underlying *co-operation mechanism* devised by the system's designer. Generally, co-operation mechanisms rely on the agents' ability to communicate with one another to exchange information relating to plans, goals and synchronisation [60]. The main techniques available to the designer to implement a co-operation mechanism are the introduction of certain control structures and communication protocols in the system, the specification of agents' goals and actions and the definition of interactions among the agents' actions. Regardless of how it is actually implemented, the co-operation mechanism of a multi-agent system provides the means to support the co-operative actions allowed by the characteristics of the agents in the system.

Two main approaches to the concept of co-operation in multi-agent systems have been identified from a review of relevant research literature: the *behaviouristic* approach and the *reflective* approach. The behaviouristic approach considers co-operation as a property of the agents' actions that results in a common benefit for the entire system, from an external observer's perspective. This approach endorses a view of co-operation that overlooks the agents' internal structures and concentrates on the agents' actions. On the other hand, the reflective approach adopts the intentional stance towards agency and supports a view of co-operation based on the internal states of the agents involved. According to the reflective approach, co-operation occurs when agents are willing to co-operate and are committed to act together for a common purpose. These two approaches to co-operation are explained in detail below. The explanations include a description of co-operation mechanisms from the multi-agent systems literature that implement each approach.

2.3.1 Co-operation in multi-agent systems: the behaviouristic approach

According to the behaviouristic approach, agents in a multi-agent system co-operate when they actually help each other with their activities in the environment where they are deployed [66]. Co-operation is regarded as a property of the agents' actions and occurs when these actions satisfy either or both of the following conditions [68]:

- The agents perform actions that achieve not only their own goals, but also the goals of other agents in the multi-agent system. Goals can either be *explicitly* known to the agent or be *implicitly* encoded as a set behaviour designed in such a way that the agent can be seen as pursuing the achievement of a goal.

- The agents share a goal that no agent can achieve on its own and the agents' actions contribute to the achievement of that goal. Again, this goal can either be an explicit goal known to the agents or an implicit goal imposed on the agents' individual behaviours by the designer.

Therefore, co-operation does not require agents to intend to co-operate for co-operative behaviour to arise. Co-operation occurs when an external observer identifies that agents are helping each other in a co-ordinated manner, regardless of whether this co-operation simply emerges from the agents' independent actions or is the result of the agents' explicit intentions to co-operate. The behaviouristic approach represents an all-inclusive view of co-operation, which focuses on the agents' actions rather that on their internal structure.

The behaviouristic approach to co-operation is implemented via behaviouristic co-operation mechanisms, which can be classified into two main types:

- **Implicit co-operation mechanisms**, which provide multi-agent systems, comprised of reactive agents, with the means to achieve co-operative behaviour. These co-operation mechanisms take advantage of the agents' actions towards the environment and each other to achieve co-operation. Thus, co-operation emerges from the interaction of the agents' individual behaviours.

- **Knowledge-based co-operation mechanisms**, which support co-operative behaviour in multi-agent systems comprised of knowledge-based agents. The agents in the system are provided with comprehensive understanding of the co-operation mechanism so that they know how to co-operate. This knowledge enables them to exchange information related to their goals and to their actions with the objective of achieving co-operative behaviour.

2.3.1.1 Literature review of implicit co-operation mechanisms

Implicit co-operation mechanisms provide a distributed control structure for multi-agent systems comprised of reactive agents. Even though individual reactive agents are merely capable of responding to their perceptions and performing basic pre-defined tasks, they are able to display sophisticated co-operative behaviours, without the need of a complex central controller, through the implementation of implicit co-operation mechanisms. These co-operation mechanisms are usually designed specifically for a particular multi-agent system comprised of homogeneous agents, which are endowed with defined capabilities and situated in a specific environment.

Generally, reactive agents are unable to exchange information with one another. However, they are often capable of communicating implicitly through the broadcasting of signals that can be received by other agents or through the mere performance of actions whose effects are perceivable by other agents. Implicit co-operation mechanisms usually consist of pre-defined responses of individual agents to stimuli perceived through implicit communication. The combination of the agents' responses to specific actions performed by other agents and to particular changes in their environment caused by other agents' actions may appear as coherent co-operative behaviour to an external observer. Some examples of implicit co-operation mechanisms are described below.

In [70] a co-operation mechanism for a robotic soccer team is presented. The mobile robots forming the team are reactive agents capable of performing basic behaviours such as obstacle avoidance and ball handling. The co-operation mechanism is encoded in the robots' responses to changes in the environment. Co-operation emerges from the interactions between the robots and the environment. Even though the robots lack an internal model of their environment and are unable to communicate explicitly, the team displays sophisticated co-operative behaviours such as the dynamic configuration into offensive and defensive formations.

An implicit co-operation mechanism based on the broadcasting of signals is presented in [71]. The co-operation mechanism enables a group of reactive mobile robots to perform synchronised actions such as lifting or steering. The co-operation mechanism is based on the ability of each robot to broadcast a unique signal, which is referred to as the robot's heartbeat. These heartbeats are perceivable by all the robots in the group. The co-operation mechanism lies in the robots' pre-defined responses to the heartbeats they perceive.

Some implicit co-operation mechanisms are inspired by the collective behaviour of social insects such as ants. Ants have a very limited memory and their individual behaviours appear to have a large random component. However, they are capable of establishing a short route from their colony to a feeding source with no initial knowledge of the source's location. If an ant, wandering aimlessly in an unknown environment, finds a feeding source, it will seize some food and head back to its colony. As it returns, it will leave behind a trail of volatile chemicals called pheromones. The pheromones attract other ants, which follow the chemical trial to the feeding source. Initially, these ants may take different routes back to the colony, along which they release pheromones that can in turn be tracked by other ants to reach the source. As more ants reach the feeding source and return to the colony, a pheromone trial is reinforced along one of the shorter routes, which is more likely to be followed by the ants to reach the feeding source and return to the colony before the pheromones diffuse. Eventually, all the ants follow the short route to reach the source and return to the colony. Hence, an efficient guiding path from the colony to the feeding source is constructed without the individual ants actually having any knowledge about how to reach it.

Ants can be seen as reactive agents and their collective foraging behaviour can be analysed adopting the behaviouristic approach to co-operation in multi-agent systems. Co-operative behaviour emerges from the ants' basic actions through a co-operation mechanism based on the release and detection of pheromones. Co-operation mechanisms, directly based on the ants' collective foraging behaviour, have been implemented in multi-agent systems comprised of reactive software agents embedded in a telecommunication network [72]. A telecommunication network interconnects a set of nodes and supports calls between arbitrary nodes. When a node is managing an excess of information it can become overloaded, causing calls to be lost. The software agents are intended to contribute to the load balancing in the network. They move across the network emulating the ants' foraging behaviour. As they move from node to node, they release simulated pheromones. The calls supported by the network are routed as a function of the pheromone distribution in the different nodes.

In [73] a group of mobile robots are provided with an implicit co-operation mechanism based on the ants' collective behaviour that enables them to co-operatively clean the floor of a building. The robots are capable of leaving chemical odour traces perceivable by other robots as they move around the building. These traces evaporate with time. The co-operation mechanism is based on the release and detection of these volatile traces and includes the algorithms that define the robots' individual behaviours according to their different perceptions. The combined actions of all the robots result in the exhaustive cleaning of the floor and appear as collective co-operative behaviour to the external observer.

The work described in [74] introduces an implicit co-operation mechanism for reactive mobile robots handling material in a workshop. This co-operation mechanism is also inspired by the organisational principles of the ants' foraging behaviour. The robots must load pieces of material from given locations in the workshop and transport them to other locations avoiding possible obstacles. Both loading and unloading points incorporate a beacon. The beacons emit a signal that attracts the robots. The robots are also equipped with beacons. The co-operation mechanism consists of the definition of the behaviour of the robot regarding the beacons. Thus, when an agent detects a beacon signal, it starts searching for that beacon. Simultaneously, the robot impersonates the beacon it is searching for by re-emitting the beacon signal. This behaviour attracts other robots that follow him in the search. This simple co-operation mechanism gives rise to interesting collective behaviours and improves the global performance of the robots in the material handling process.

2.3.1.2 Literature review of knowledge-based co-operation mechanisms

Knowledge-based co-operation mechanisms provide a means by which co-operative behaviour can be achieved in multi-agent systems comprised of knowledge-based agents. The co-operation mechanism lies in the agents' comprehensive understanding of the methods and procedures that lead to co-operative behaviour. Thus, each agent is assumed to have sufficient knowledge about how to co-operate to ensure that both its individual goals and the goals of the other agents in the system are achieved in a co-ordinated manner. Even though the agents must know which actions to perform in order to act co-operatively, they do not need to know why those particular actions are the appropriate ones to be performed. The co-operation mechanism ensures that each individual agent chooses to perform actions that lead to co-operative behaviour, irrespectively of the agent's criteria to select those actions. Therefore, knowledge-based co-operation mechanisms enable the agents to act co-operatively without the need for them to specifically elaborate and agree upon co-operation strategies. Knowledge-based co-operation mechanisms usually rely upon the exchange of information between agents, which results in a global system behaviour that can be seen as co-operative by an external observer. Some examples of knowledge-based co-operation mechanisms are described below.

In [75] a knowledge-based co-operation mechanism for a multi-agent system comprised of planning robots is presented. The mechanism relies on the ability of the robots to exchange information about their individual plans. The robots take into account the information received from the other robots in the system to amend their individual plans so that their actions do not conflict. From the point of view of an external observer the agents appear to behave co-operatively as the global system performance is improved through the prevention of conflicts between the agents.

The co-operation mechanism presented in [76] models human co-ordination procedures used in military aviation. The mechanism is designed for a multi-agent system comprised of knowledge-based agents that simulate pilots and controllers involved in a military mission. Each agent's actions are determined by a set of internal rules that are applied whenever their conditions are met. The agents are capable of exchanging explicit information through simulated radio transmissions. The co-operation mechanism lies in the agents' explicit knowledge of the co-operation methods and procedures. To successfully accomplish a mission, the agents may have to communicate with each other to organise themselves into groups with different goals. The agents have internal representations of the possible organisational structures in which agents can participate and they also posses extensive knowledge about the different roles that they can perform within the organisational structures. Therefore, the agents know how to co-operate so that successful global behaviour is achieved in any given situation. However, they do not know why they have to co-operate. They do not even intend to co-operate. The agents simply apply their internal rules according to their perceptions and their knowledge.

The work described in [77] presents a co-operation mechanism for a multi-agent system comprised of simple knowledge-based agents that do not have a model of the other agent's plans or goals, nor are they capable of communicating. Nevertheless, the agents know the current state of their environment and decide which actions to perform according to this information. The co-operation mechanism is incorporated into a rule for sociable behaviour and added to the agents' decision-making process. This rule induces agents to carry out extra actions in the environment. These actions will indirectly facilitate the actions of other agents. Therefore, co-operation arises from unilateral co-operative actions performed by individual agents. The agents know the additional rule that embodies the co-operation mechanism but they do not know that by applying that rule they are actually contributing to the improvement of the global system performance through co-operative behaviour.

The Recursive Modelling Method (RMM) was proposed in [78] to enable the co-ordination of the actions of non-deliberative planning knowledge-based agents in the framework of a multi-agent system. Each agent in the system is assumed to select the course of action that maximises its *expected utility*. The expected utility is a function that maps a goal onto a real number. This number describes the degree to which the agent is satisfied with the achievement of the goal. Thus, the utility of a goal represents the difference between the benefit yielded as a result of the achievement of that goal and the cost of the actions performed by the agent to achieve the goal. The RMM is specifically designed for cases where the agents are not able to communicate explicitly with each other and there are no pre-defined protocols or conventions about how they should interact. According to the RMM, the agents must consider their knowledge about the other agents' plans and goals when they assess the potential outcomes of their own actions. Hence, each agent elaborates an internal model of the other agents in the multi-agent system. This model allows the agents to predict the other agents' actions with a degree of uncertainty. Each agent considers the actions of the others as it chooses its own course of action. When an agent is modelling other agents it may consider that those agents are similarly modelling it as they plan their actions. The agent may also consider that the other agents model how it models them, and so on. This reciprocity leads to a recursive nesting of models, which is assumed to be finite due to the practical impossibility where an infinite nested knowledge is achieved. The Recursive Modelling Method endows the agents with this nested representation of the other agents in the system. Based on this internal representation of the other agents, each agent applies a dynamic programming algorithm to select a course of action that maximises its expected utility. The agent's utility function depends on the uncertainties associated with the agent's predictions of the other agents' possible actions. The algorithm provides a preferred course of action that, to the best of the agent's knowledge, will not conflict with the potential actions of the other agents.

In cases where the agents benefit from the achievement of one another's goals, the Recursive Modelling Method gives rise to seemingly co-operative behaviour. The agents appear to help one another achieve their goals, even though they are self-interested and pursue their maximum individual utility. Each agent considers its knowledge about the other agent's goals and possible actions as it searches for a course of action that maximises its utility. This course of action is likely to be one that facilitates the other agents' actions in the pursuit of their goals, since the attainment of the other agents' goals would yield an increase in the agent's own utility function.

2.3.2 Co-operation in multi-agent systems: the reflective approach

While the behaviouristic approach considers co-operation as a property of the agents' actions, the reflective one focuses on the agents' mental states that may lead them to co-operate with one another. Consequently, the reflective approach implies that co-operation is only possible among agents designed as intentional systems. According to this approach, a group of agents are deemed to co-operate when they are in a specific mental state that drives them to act together, towards the achievement of a *common goal*, irrespectively of whether or not they are seen as helpful to one another by an external observer ([66], [68]). A common goal is shared by all the agents in the group and is liable to be achieved through joint activity. Consider the following example as an illustration of the concept of common goal. Suppose two agents, A and B, want to cook a haggis. They could decide to work as a team and cook the haggis together. If they did so, the two agents would have the common goal of cooking the haggis and would be co-operating to achieve this goal. Suppose now that the haggis has been cooked and both agents wish to eat it. In this case, they share a goal that cannot be attained by them acting together. Instead, the agents compete with each other to achieve their goal as both of them attempt to eat the same haggis.

Since all share the same common goals, the agents individually prefer to achieve those goals. Hence, co-operation towards a common goal can be seen as partially self-interested. On the other hand, the agents are willing to act together towards that goal, showing an element of mutual helpfulness. To ensure that the achievement of a common goal is the result of the agents consciously engaging in joint activity, rather than of accidental co-ordination, it is assumed that the agents in the group are aware that they all share the goal. To illustrate accidental co-ordination, consider the following example. Both agents A and B want again to cook a haggis but, in this case, the two agents are unaware of each other's goal. Agent A decides to postpone the cooking until later, while agent B decides to start cooking immediately. When agent A decides to start cooking, it finds that the haggis has already been cooked by agent B. Thus, one agent achieved the goal of both of them and yet was unaware of the other's goal. This is an example of accidental co-ordination, which is not considered co-operation according to the reflective approach.

2.3.2.1 Literature review of reflective co-operation mechanisms

In general, co-operation mechanisms designed to implement the reflective approach to co-operation support the formation of *teams*. A team is comprised of agents that share a common goal and are committed to perform a *joint action* to achieve that common goal. The joint action enables the team members to support one another by closely co-ordinating their individual actions.

Since the agents are designed according to the intentional stance, they are ascribed mental states that emulate human mental attributes. The mental states, which are implemented in the agent program, are responsible for guiding the agent's individual behaviour as well as its social behaviour. The basic mental state is *belief*. The beliefs of an agent refer to the pieces of knowledge about which the agent is aware. Other important mental states include *commitments* and *conventions*, which are considered essential in the generation of joint actions [66]. A commitment generally refers to an agent's pledge to undertake a specified course of action. An agent can also commit to the achievement of one particular goal, irrespectively of the actions it will perform towards the attainment of that goal. Agents must endeavour to honour their commitments. Conventions are policies applied by an agent to govern the reassessment of its current commitments. They describe situations under which the agent should reconsider its commitments and, when such situations arise, they indicate whether the commitments have to be retained, rectified or abandoned. The conventions also establish how an agent should act, both locally and towards its fellow team members, when it alters its commitments.

Some examples of reflective co-operation mechanisms are briefly described below.

The reflective co-operation mechanism presented in [66] focuses on the characterisation of the mental states needed to support joint actions in multi-agent systems designed for industrial applications. These systems are comprised of agents that have neither complete nor correct beliefs about their environment or about the other agents in the system. The co-operation mechanism relies on the agents' ability to communicate explicitly with each other and is based on a mental state called *joint responsibility*. The mental state of joint responsibility must be adopted by all the agents in a team if they are to perform a joint action, aiming to achieve a common goal. To achieve the mental state of joint responsibility, all the team members must commit to the undertaking of a common plan towards the attainment of that common goal. In addition, each of them must believe that the other members of the team share that same commitment. The mental state of joint responsibility involves a set of conventions that guarantee coherent team activity. A *modal dynamic logic* ([66], [61], [79]) is used to describe the mental state of joint responsibility formally. This logic provides the means to describe precisely the specifications for the co-operation mechanism. Once the mental state of joint responsibility has been formalised, a computational model to support the implementation of the co-operation mechanism is devised. In this computational model, the behaviour of the individual agents is guided by a set of rules. These rules implement the mental states needed to form teams and perform joint actions. The processes of forming a team and establishing a joint action according to this co-operation mechanism are described next.

The team formation process commences when an agent recognises the need for a joint action to achieve a certain goal and decides to form a team. This agent is called *team organiser* and contacts the agents it believes able to co-operate. Subsequently, a protocol encoded in the agents' internal rules and based on the mental state of joint responsibility, guides the communications between the team organiser and those agents, some of which adopt the goal proposed by the team organiser and acknowledge the need for a joint action to achieve it. The agents that decide to co-operate, together with the team organiser, have achieved the mental state of joint responsibility and constitute a team. Once the team has been formed, the team organiser elaborates a common plan and assigns to the team members their actions within the plan. If the agents consider that the actions they have been assigned are feasible according to their skills and knowledge, they send a message to the team organiser indicating their acceptance. Due to the team organiser's incomplete knowledge of the environment and the other agents in the team, it might propose actions that are unattainable by its team mates. If an agent is unable to perform the action that the team organiser has allocated to it, it searches for an alternative feasible action that it believes contributes to the achievement of the common goal and suggests it to the team organiser. If the team organiser considers it acceptable, it makes the appropriate adjustments to the common plan and informs its fellow team members of these adjustments. If the new action proposal is unacceptable, the team organiser searches for another agent within the team willing to perform the original action. Once all the actions within the common plan have been established and accepted, the organiser sends out a message to all the team members indicating that a final agreement has been achieved and the execution of the common plan commences.

The work presented in [80] investigates the application of the co-operation mechanism described above to a simulated pursuit game. A simulation shows how pursuer agents work in teams to catch individual targets.

The *contract net* [81] provides agents in a multi-agent system with the means to associate and co-ordinate their actions, the objective being to execute a complex task. The contract net focuses on the distribution and co-ordination of subtasks among a group of agents that aims to share the workload created by a complex task. The agents are capable of communicating with one another and are willing to co-operate. The contract net is a communication protocol that determines the information to be exchanged by the agents to achieve co-operative behaviour. The protocol is based on the concept of *negotiation*, which is understood to be a "discussion" in which interested parties exchange information and come to an agreement. In the contract net, when an agent recognises that a complex task must be performed, it divides the task into manageable subtasks and announces these subtasks to the other agents in the system. The agent that announces the subtasks is called the *manager*. The recipients of the announcement evaluate the subtasks according to their interests and resources. If an agent decides to commit to performing a subtask, it submits a bid to the manager. The manager may receive several bids for the same subtask. The bids contain information about the capabilities and interests of the bidders. Based on that information, the manager selects one or more agents to execute the subtask. Thus, the manager has established a contract with the successful bidders, which are called *contractors*. Contractors may in turn divide a subtask and establish subcontracts with other agents. The contract net protocol can be considered as a reflective co-operation mechanism that enables agents to form teams through contracts. The agents involved in a contract are committed to act together towards the achievement of a common goal: the execution of the task at hand.

The co-operative mechanism presented in [82] provides a teamwork structure for multi-agent systems situated in environments where the agents alternate between periods of restricted and unrestricted communication. The mechanism has been successfully applied in computer-simulated robotic soccer, where team members can plan strategies before the game and at halftime, but communication is limited during the course of the match. According to this co-operation mechanism, each team member adopts a *role*, which is a function of its internal mental states and determines its behaviour. In the context of computer-simulated robotic soccer, the agents' roles are their positions on the field, such as goalkeeper, centre back or right midfielder. The agents are supplied with the knowledge necessary to switch between different roles. The team can be organised into several different *formations*. The formation, which defines the strategy of the team, is established by assigning each team member a role. For example, consider a team of eleven agents in the domain of computer-simulated robotic soccer. A possible formation could be one goalkeeper, four defenders, four midfielders and two strikers. The definitions of all the possible roles and formations are known to all agents. During the periods of unrestricted communication, the team members can exchange information freely and agree on a particular formation depending on the task at hand. During the periods of limited communication, a protocol ensures that the agents can inform each other of their roles and formations periodically. Thus, if one team member decides to change the formation, it can communicate its decision to its team mates, which eventually switch to their new role according to the new formation.

The work presented in [83] introduces a reflective co-operation mechanism based on providing all the team members with exhaustive knowledge of the team's goals, plans and organisational hierarchy as well as with an explicit model of teamwork. The team members exchange information according to this model to achieve the mental states that lead to joint activity. This co-operation mechanism has also been successfully applied in computer-simulated robotic soccer.

2.4 Multi-agent-based model of Autonomous Aircraft Operations

The remainder of this thesis will explore the potential application of concepts and techniques from multi-agent systems to develop co-operative conflict resolution methodologies for Autonomous Aircraft Operations. To do so, a multi-agent-based model of the operational concept for AAO adopted in this thesis is proposed. In this model, Autonomous Aircraft are modelled as agents and conflict resolution in AAO is analysed in the context of a multi-agent system.

2.4.1 Autonomous Aircraft as knowledge-based agents

Autonomous Aircraft are modelled as knowledge-based agents whose environment is the volume of airspace within which Autonomous Aircraft Operations take place. The aircraft-agents are able to interact with their respective flight crews as well as with the other aircraft-agents in the vicinity. The interactions among aircraft-agents are considered in the context of a multi-agent system. This new multi-agent approach to AAO is intended as a conceptual framework within which conflict resolution in AAO can be analysed. The analysis will allow for the development of co-operation mechanisms that make co-operative conflict resolution possible.

The aircraft-agents explicitly know about their environment, about themselves and about other agents and are able to decide which actions they should undertake to achieve their goals. They are capable of acquiring additional knowledge through communication with other agents, with the ground and with their respective pilots. These communications take place in different forms. The communication between the aircraft and the flight crew uses a computer-like human-machine interface, which, amongst other functions, displays information relating to the aircraft's possible courses of action and interprets the flight crew's instructions. Communication with the other aircraft-agents and with the ground consists of the transmission and reception of information through a data-link.

The aircraft-agents' knowledge is modelled as a conceptual knowledge base, which is an abstraction of the information stored in the aircraft's avionics systems. The knowledge base comprises of what the agent knows about its environment, its proximate aircraft and itself, including its knowledge relating to its performance capabilities and its goals. The knowledge base is constantly updated with new pieces of information acquired through communication and through the agent's sensors as well as with information obtained by its interaction with the pilot.

The aircraft-agents assist their respective flight crews in maintaining safe separation with the proximate Autonomous Aircraft. Each aircraft-agent exchanges information with the surrounding agents and uses this information to provide the flight crew with adequate decision support to resolve conflicts in a co-operative and efficient manner. The conflict resolution process is governed by a co-operation mechanism that aims to ensure that the resolution manoeuvres are co-ordinated and that the conflict resolution costs are shared equitably by all the aircraft involved. The aircraft-agents are also capable of calculating and flying airline-preferred routes. In addition, they are able to dynamically modify their current route in an efficient way to avoid hazardous weather conditions or comply with flow management restrictions. The aircraft-agents have the required knowledge and capability to acquire timely information from their environment in order to perform these activities. Thus, they have knowledge about their flight performance as well as about the airline's preferences regarding fuel consumption and flight times. They are considered capable of perceiving their environment through sensors that provide them with wind and weather information. They can obtain additional meteorological information through communications with ground stations. Information relating to Air Traffic Flow Management can also be received from ground-based ATFM services.

Separation assurance occupies the highest priority in the aircraft-agents' agenda, prevailing over their drive to optimise their routes. Nevertheless, although the aircraft-agents' main goal is to assist their respective flight crews in flying conflict-free trajectories, they take into account flight efficiency as well as safety when planning conflict resolution manoeuvres.

The knowledge-based agent model of Autonomous Aircraft presented above is intended to serve as a tool to analyse airborne separation assurance in AAO in the context of a multi-agent system. This analysis is expected to provide ideas for co-operation mechanisms that support co-operative conflict resolution. To prevent these co-operation mechanisms from relying on a specific internal logic used by the aircraft-agents to represent knowledge and reason explicitly, the aircraft-agents are assumed to be non-deliberative. Thus, to apply a co-operation mechanism, the aircraft-agents use computational methods incorporated in their agent programs, which run on the airborne computer system. These computational methods operate by using the agent's knowledge gathered from the aircraft's avionics.

A diagram representing the conceptual structure of the agent program of an Autonomous Aircraft modelled as a non-deliberative knowledge-based agent is depicted in Figure 2.7.

2.4.2 Co-operation in a multi-agent system as a model for conflict resolution in Autonomous Aircraft Operations

Autonomous Aircraft are viewed as agents situated in an environment containing other agents, with which they interact to detect and resolve conflicts. These interactions

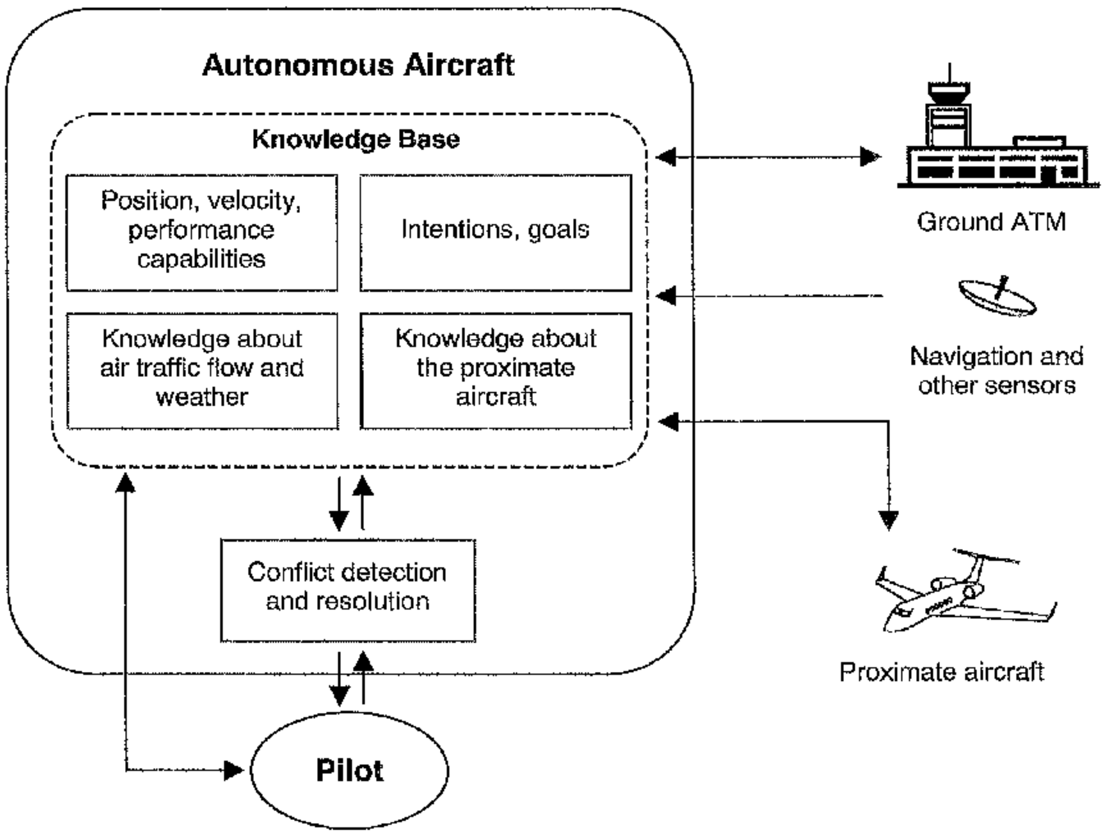


Figure 2.7: Structure of the agent model of an Autonomous Aircraft.

consist mainly of the transmission and reception of ADS-B messages. Through ADS-B surveillance, the aircraft-agents become aware of the presence of other aircraft-agents and acquire information about their position, speed and intentions. Additionally, the aircraft-agents may be able to interact with one another through the exchange of information via a point-to-point data-link. The interactions among the aircraft-agents will be considered in the context of a multi-agent system with the objective of developing methodologies for co-operative conflict resolution based on those interactions. A group of aircraft-agents conflicting with one another are viewed as constituting a multi-agent system, in the context of which they co-operate to achieve satisfactory global performance in the resolution of their conflicts. In this multi-agent approach, conflict resolution in AAO is considered an essentially co-operative process.

Co-operative conflict resolution is achieved through a co-operation mechanism. The co-operation mechanism of a multi-agent system determines the type of co-operation attainable in the system by establishing the rules, algorithms and protocols needed for the agents to co-operate. The design of a co-operation mechanism for conflict resolution in AAO is influenced by the structure and characteristics of the individual aircraft-agents as well as on the nature of their interactions.

The remainder of this thesis will concentrate on the investigation of possible co-operation mechanisms for conflict resolution in multi-agent systems of conflicting Autonomous Aircraft. Two different co-operation mechanisms will be proposed to support co-operative conflict resolution in two potential operational environments for the operational concept of AAO adopted. The mechanisms are introduced in the next section and will be explained in detail in the following chapters of this thesis.

2.4.3 Co-operation mechanisms for conflict resolution in Autonomous Aircraft Operations

In this section, the two co-operation mechanisms for conflict resolution in AAO that constitute the main contribution of this thesis will be introduced. Both co-operation mechanisms are based on the multi-agent based model of AAO presented above and are intended to illustrate the potential of this model to support co-operative conflict resolution methodologies.

Two different operational environments for the operational concept of AAO adopted are considered. Each operational environment corresponds to a different level of airborne equipment. A co-operation mechanism has been specifically designed to support co-operative conflict resolution in each of the two different operational environments. The mechanisms illustrate the two different approaches to co-operation in multi-agent systems of knowledge-based agents. Thus, one of them is a knowledge-based co-operation mechanism and implements the behaviouristic approach to co-operation, while the other is a reflective co-operation mechanism and relies on a conception of the aircraft-agents as intentional systems.

In the following chapters of this thesis, both operational environments will be described together with their corresponding co-operation mechanisms. The first operational environment considered, denoted as Operational Environment A, comprises of the minimum equipment requirements for the operational concept of AAO adopted here. The corresponding knowledge-based co-operation mechanism endorses a behaviouristic view of co-operation and is inspired by the Recursive Modelling Method (RMM). In the second operational environment considered, Operational Environment B, the aircraft are also equipped with a point-to-point data-link. This data-link enables aircraft to address one another and exchange information on a one-to-one basis. The co-operation mechanism for Operational Environment B implements a reflective approach to co-operation and is inspired by the co-operation mechanism described in [66], which is based on the concept of joint responsibility.

Chapter 3

Behaviouristic co-operation in Autonomous Aircraft Operations

3.1 Introduction

This chapter illustrates the use of concepts and techniques from multi-agent systems to develop co-operative conflict resolution methodologies for AAO in Operational Environment A. A co-operation mechanism is proposed as a co-operative conflict resolution methodology for the operational environment considered. This co-operation mechanism provides the means for conflicting Autonomous Aircraft to safely co-ordinate their resolution actions so that they share the resolution costs. The algorithms and procedures that constitute the co-operation mechanism are designed so that they can be integrated into the on-board ASAS equipment to guide the flight crew's decision-making during the conflict resolution process.

As explained in the previous chapter, Autonomous Aircraft are modelled as knowledge-based agents that interact with one another in the context of a multi-agent system. In Operational Environment A, aircraft-agents are capable of interacting with each other only through the broadcast and reception of ADS-B messages, as no point-to-point data-link facility is available. Thus, they are unable to address each other and exchange information to form a team and establish a common conflict resolution strategy. Consequently, the development of a co-operation mechanism based on the reflective approach to co-operation is not feasible in this operational environment. The behaviouristic approach has been adopted instead and a knowledge-based co-operation mechanism for Operational Environment A is proposed. The proposed mechanism relies on the broadcast of intent information through ADS-B and is inspired by the Recursive Modelling Method (RMM), which was described in section 2.3.

In the remainder of this chapter, Operational Environment A is described focusing on the information that the ADS-B messages must include so that the implementation of the proposed co-operation mechanism is possible. Subsequently, the proposed co-operation mechanism is explained in detail and illustrated with examples.

3.2 Operational Environment A: ADS-B-based Autonomous Aircraft Operations

Operational Environment A defines the minimum requirements to implement the AAO concept proposed in this thesis. The main features of this operational environment A are listed in Table 3.1.

The ground-based ATFM service allocates specific volumes of airspace to AAO. It is assumed that the air traffic density in the airspace designated for AAO is regulated by the ground-based ATFM service so that the number and complexity of the possible conflicts are manageable by the conflict resolution methodology in place. To accommodate forecasted changes in air traffic density, the ground-based ATM service may, for example, adjust the traffic flows entering AAO airspace or modify the boundaries of the volumes of airspace allocated for AAO.

Only suitably equipped aircraft are allowed to enter AAO airspace. Once they enter AAO airspace, these aircraft become Autonomous Aircraft. Within AAO airspace, Autonomous Aircraft fly operator-preferred routes and are fully responsible for separation assurance. ADS-B is the only means for Autonomous Aircraft to exchange

Airspace characteristics	
Airspace structure	• Airspace allocated to Autonomous Aircraft Operations. • Generally high altitude (above FL335). • No ATC coverage (possibly oceanic and over remote areas). • No fixed route structure: users fly their preferred routes between entry and exit points to Autonomous Aircraft operations airspace.
Traffic structure	• Generally en-route cruising traffic. • Possibly end-of-climb and top-of-descent traffic. • Likely crossing points between aircraft routes.
Separation assurance criteria	• As described in Chapter 1, Section 1.3.
Air Traffic Management support	• No ATC support: separation assurance responsibilities fully delegated to the flight crew. • ATFM service: air traffic density has to be maintained below a certain level so that conflicts are manageable by ASAS.
Aircraft Equipment	
Communications	• Air-ground and air-air voice communications. • Air-ground data-link: exchange of information with the ground ATM centres.
Navigation	• GNSS equipment. Fully airborne navigation (no ground-based navigation aids).
Surveillance	• ADS-B equipment.
Traffic Situational Awareness and Separation Assurance	• Appropriate ASAS equipment: conflict detection and resolution support tools. • Pilot-ASAS interface, including CDTI and ASAS control panel.
Others	• 4D-FMS • ACAS mandatory.

Table 3.1: Schematic description of Operational Environment A.

information with one another. Therefore, airborne conflict detection and resolution is based on data broadcast through ADS-B.

A voice link is available for the Autonomous Aircraft's pilots to communicate with each other. It is supposed that the ASAS equipment supports conflict resolution without the need for the pilots to talk to each other. Voice communications are to be used only in the event of a non-nominal occurrence, such as emergency or equipment malfunction.

Regarding ADS-B, no assumption is made about which particular ADS-B data-link technology is used. The three main candidate data-link technologies to support ADS-B are explained in Appendix B. The decision about which technology will be implemented and certified as the standard ADS-B data-link is still to be made by the competent international aviation institutions. For the purpose of the research described in this thesis, it is assumed that Autonomous Aircraft are capable of periodically broadcasting messages containing information regarding their identity, position, speed and intent, irrespective of the ADS-B data-link technology in place. The ADS-B equipment enables aircraft to encode and decode the information contained in the ADS-B messages. The ADS-B messages are received by all the aircraft flying within a certain range of coverage that will depend on the data-link in place. It is expected that the ADS-B range of coverage will reach up to 120 nm.

3.2.1 ADS-B messages

The minimum information to be included in the ADS-B messages to support the proposed AAO concept is outlined below:

- Aircraft identification, which is the ICAO airframe address.

- Aircraft three-dimensional position, together with the time at which the position is applicable.

- Aircraft ground speed, together with the time at which the speed is applicable.

- Aircraft intent, which is information describing the planned route. The aircraft intent is obtained from the 4D-FMS and refers to the trajectory that is to be flown by the aircraft. In addition to the intent information, the time at which the pilot selected the current intent is also included in the ADS-B messages. Furthermore, when the pilot decides to modify the aircraft's currently intended trajectory, a flag announcing an intent change is included in the ADS-B message.

- Special operational status, such as emergency or high priority.

The broadcast of the exact times at which the information contained in the message is applicable is intended to contribute to mitigate the effects of possible uncertainties and ambiguities caused by the delays involved in the processes of encoding, transmitting and decoding the information. It is assumed that the ADS-B data-link technology in place will prevent the simultaneous transmission of information by more than one user. Therefore, no more than one aircraft can transmit data at one time and the loss of information due to the interference caused by the overlapping of two ADS-B transmissions is avoided.

The message transmission rate will depend on the data-link in place, on the length of the message and on the number of aircraft using the data-link frequency. For the purpose of this research, it is assumed that each Autonomous Aircraft receives sufficient information from its surrounding Autonomous Aircraft to keep a reliable and updated picture of its surrounding traffic. It may occur that some parts of the ADS-B message are broadcast at different rates than others. Information regarding aircraft identification and intent may be broadcast at slower rates than aircraft position and speed. However, changes in the aircraft intent route must be broadcast immediately to enable the surrounding aircraft to detect possible conflicts created by the new trajectory.

3.2.1.1 Aircraft intent

It is assumed that the aircraft intent information encodes the aircraft's four-dimensional planned route to its exit point of AAO airspace. This provides the flight crews with a picture of the evolution of their surrounding traffic within the airspace where separation assurance is delegated to the cockpit, allowing for the detection of conflicts long before the closest point of approach. Early conflict detection enables Autonomous Aircraft to plan conflict resolution actions taking into account cost efficiency.

Aircraft intent information is encoded using a *trajectory language* understandable by all Autonomous Aircraft and capable of describing the aircraft's planned four-dimensional trajectory with sufficient accuracy. The work described in [84] can be considered as an important milestone in the development of such a language, inasmuch as it introduces a simple and structured *path language* that could be used by the FMS to encode three-dimensional aircraft trajectories.

The separation minima in place will depend on the exactness and completeness of the intent information as well as on the aircraft's accuracy to fly their intended route. Therefore, it may be required that information regarding the completeness and accuracy of the intent data and the tracking capabilities of the aircraft is incorporated into the ADS-B messages. The ASAS equipment would use this information to establish the sizes of the Protection and Monitoring Volumes.

3.3 Co-operative conflict resolution for AAO in Operational Environment A

The behaviouristic approach to co-operation has been adopted to develop a co-operation mechanism for AAO in Operational Environment A. According to the behaviouristic approach, co-operation in a multi-agent system is achieved when the agents appear to act co-operatively, regardless of whether or not they are knowingly engaged in joint activity. In multi-agent systems comprised of knowledge-based agents, the behaviouristic approach to co-operation is implemented through knowledge-based co-operation mechanisms, which are based on the agents' understanding of the methods and procedures that lead to co-operative behaviour. The remainder of this chapter presents a knowledge-based co-operation mechanism for AAO in Operational Environment A inspired by the Recursive Modelling Method (RMM).

3.3.1 Knowledge-based co-operation mechanism for conflict resolution in Operational Environment A

This section explains how the proposed co-operation mechanism would operate in a generic multi-aircraft conflict scenario. The generic scenario considered involves n aircraft-agents located within the ADS-B coverage range of all other aircraft, so that each aircraft is able to detect all the other n-1 aircraft. It is assumed that each aircraft is in conflict with at least one other aircraft in the scenario.

According to the agent model of Autonomous Aircraft adopted in this thesis, aircraft-agents aim to resolve their conflicts in a co-operative manner. Due to the absence of a point-to-point data-link facility in the operational environment considered, conflicting aircraft-agents are unable to establish a common resolution strategy. Therefore, co-operative conflict resolution is to emerge from the conflicting aircraft's individual resolution actions. These resolution actions are generated according to the knowledge-based co-operation mechanism in place, which is known to all the Autonomous Aircraft.

Each aircraft-agent models its conflicting aircraft considering that they are willing to collaborate towards the resolution of their conflicts. The co-operation mechanism establishes that each conflicting aircraft-agent plans its resolution actions taking into account the possible resolution actions of the aircraft-agents with which it is in conflict. When an aircraft-agent models other aircraft-agents to anticipate their actions, it may take into account the fact that those aircraft-agents are also modelling it to anticipate its actions. If it does so, the aircraft-agent has to model how the other conflicting aircraft are modelling it. Further, the aircraft-agent could also consider that the other conflicting aircraft are also modelling how they are been modelled by it. This recursive modelling could continue on to the aircraft-agent modelling how the other conflicting aircraft are modelling how it is modelling how the other conflicting aircraft are modelling it, and so on. To avoid this infinitely nested modelling structure, which is unattainable with the aircraft-agents' finite knowledge capacity, the co-operation mechanism establishes that each conflicting aircraft plans its resolution action assuming that it will act first and the others conflicting aircraft will respond to its action. This hierarchical structure of the conflict resolution process is borrowed from the theory of

Stackelberg games [85]. In such games one player, called the *leader*, declares its strategy first and enforces it on the other players, called *followers*. Among the followers there can be others levels of hierarchy, with players acting simultaneously as followers of the leader and leaders of other followers. To provide some insight into this hierarchical decision-making process, two-player Stackelberg games are briefly explained below.

3.3.1.1 Two-player Stackelberg games

In a two-player Stackelberg game one of the players is the leader and acts first, while the other responds to the strategy chosen by the leader. Each of the two players pursues its own interests, which are encoded in a cost function. The cost function of a player is influenced by the strategies of both players. It is assumed that each player knows the other player's cost function as well as the influence of the other player's possible strategies on its own cost function. The leader takes into account the possible responses of the follower to its strategies and their effect on its cost function to select the strategy that is most favourable to its interests. It is assumed that the leader attempts to secure its possible losses against the choices of the follower. Thus, it selects a strategy that produces the minimum cost assuming the most adverse response of the follower. Such a strategy is called a *Stackelberg equilibrium strategy*. A precise definition of this concept is presented next. For a generic two-player Stackelberg game, let S^1 and S^2 denote the strategy spaces of Player 1 and Player 2, respectively, and $J^i(s^1, s^2)$ denote the cost incurred to Player i corresponding to a strategy pair ($s^1 \in S^1$, $s^2 \in S^2$). Assuming that Player 1 is the leader, then the *optimal response set* or *rational reaction set* of Player 2 to the strategy $s^1 \in S^1$ of Player 1, which is denoted as $R^2(s^1)$, is defined as follows:

$$R^2(s^1) = \{\xi \in S^2 : J^2(s^1, \xi) \leq J^2(s^1, s^2), \ \ \forall s^2 \in S^2\} \tag{3.1}$$

A strategy $s^{1*} \in S^1$ is called a Stackelberg equilibrium strategy for the leader if:

$$\max_{s^2 \in R^2(s^{1*})} J^1(s^{1*}, s^2) = \min_{s^1 \in J^1} \max_{s^2 \in R^2(s^1)} J^1(s^1, s^2) \triangleq J^{1*} \tag{3.2}$$

If Player 2 is the leader, the same definition applies with only the superscripts 1 and 2 interchanged. The quantity J^{1*} in (3.2) is the *Stackelberg cost* of the leader. Once the leader has announced its chosen Stackelberg equilibrium strategy, the follower responds by searching for a strategy that minimises its cost assuming the leader's selected strategy.

3.3.1.2 Conflict resolution process defined by the co-operation mechanism

The proposed co-operation mechanism for AAO in Operational Environment A borrows some elements from the hierarchical decision-making process that is at the core of Stackelberg games. In particular, the leader-follower structure has been introduced to avoid the infinite nested modelling structure introduced by the RMM and to provide the means for the co-ordination of the conflict resolution actions. When planning a resolution action, each aircraft regards itself as the *leader* and considers the remaining conflicting aircraft as *followers*. The leader assumes that it will act first in the conflict resolution and the followers will respond to its selected action.

In a first stage of the conflict resolution process all conflicting aircraft-agents simultaneously regard themselves as leaders and consider the other conflicting aircraft-agents as followers. When searching for suitable resolution actions the leaders take into account the possible responses of the followers to those actions. To achieve co-operative conflict resolution, the resolution actions are chosen so that they facilitate the reactions of the followers. Once a resolution action has been chosen, it is proposed to the flight crew as the new intended trajectory. In nominal operation, the crews are assumed to accept the proposed resolution trajectories after a certain time lag. This elapsed time between the display of the resolution trajectory and its acceptance will depend on factors such as the crew's familiarity with the ASAS equipment and the complexity of the conflict and the proposed resolution trajectory.

The aircraft-agent whose crew first accepts the proposed resolution trajectory effectively acts as the leader and proceeds to implement that resolution trajectory. Once the crew has accepted the proposed trajectory, the new aircraft intent is broadcast via ADS-B. When the other conflicting aircraft-agents receive the leader's new intended trajectory, they withdraw from their role of leader, assume the role of follower and plan a response to the leader's new intended trajectory. As stated in section 3.2, it is assumed that the ADS-B data-link technology in place will not allow more than one aircraft to transmit data simultaneously. Therefore, in the case of two or more crews accepting the proposed resolution trajectory simultaneously or almost simultaneously, the aircraft whose new intended trajectory is broadcast first does actually implement that trajectory as the leader. Once they receive the leader's new intent, the other conflicting aircraft-agents inform their crew of the new situation and start planning a response to the leader's new trajectory assuming the role of follower.

When a follower plans its resolution action, it regards itself as the leader of the remaining conflicting aircraft as well as the follower of the leader. Thus, each follower plans its resolution trajectory assuming that it will react first to the leader's resolution trajectory and considering the possible responses of the remaining followers to its own resolution trajectory. As this planning takes place simultaneously on board all the followers, the follower whose new resolution trajectory is accepted by its pilot and broadcast first will effectively implement that trajectory and enforce it on the rest of the followers. This process continues until the last follower implements its reaction to the other conflicting aircraft's resolution manoeuvres. The sequence of aircraft actions is not pre-determined *a priori* but defined dynamically by the different random elapsed times between the proposal of a resolution trajectory and the acceptance of that trajectory by the crew. Figure 3.1 depicts how the sequence of actions is established in a conflict scenario involving three aircraft.

To implement the conflict resolution processes outlined above, the co-operation mechanism relies on a *trajectory-planning algorithm* installed on board all the Autonomous Aircraft. The algorithm is incorporated into each aircraft-agent's knowledge base and provides the crew with a resolution trajectory produced according to the co-operation mechanism. The trajectory-planning algorithm considers its host aircraft as a follower of the aircraft that have already broadcast their intended resolution trajectories and simultaneously as the leader of the rest of the conflicting aircraft. The algorithm, which has been designed for nominal flight operations, is explained in detail below.

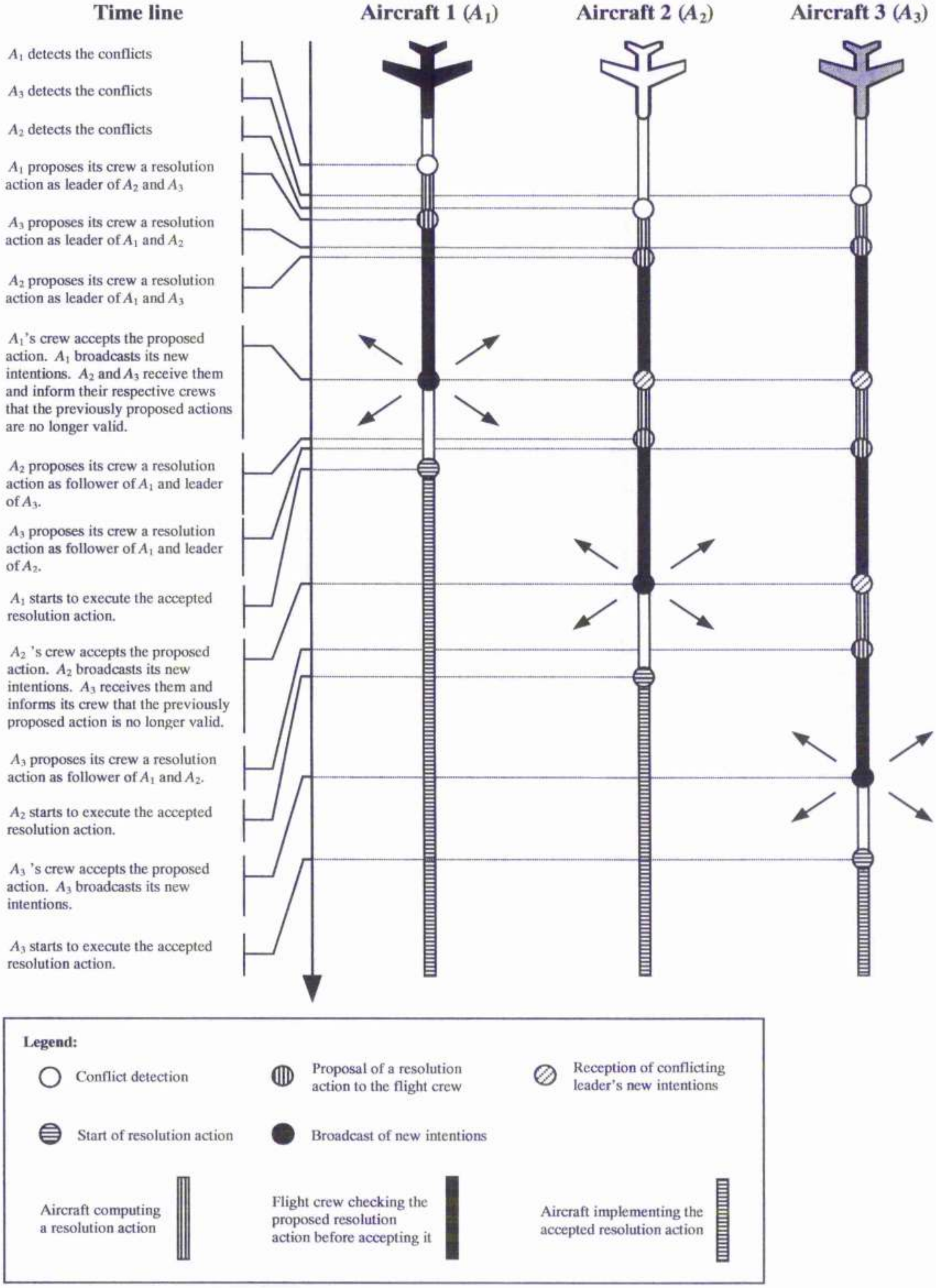


Figure 3.1: Sequence of resolution actions established by the conflict resolution methodology.

3.3.2 Operation of the trajectory-planning algorithm

The main feature of the trajectory-planning algorithm is its capability to anticipate the possible resolution trajectories that the other conflicting aircraft may undertake as a response to the resolution trajectory produced by the algorithm. To anticipate these trajectories, the algorithm relies on a finite set of *conflict resolution patterns* known to all the Autonomous Aircraft. A conflict resolution pattern is a pre-defined trajectory that is chosen to represent the set of all the conflict resolution trajectories that can be classified into the same category. For instance, a conflict resolution pattern can be defined to represent the set of resolution trajectories involving only speed control actions in conflicts between aircraft cruising along straight paths at the same altitude. An example of a conflict resolution pattern characterising such trajectories is the trajectory that results from accelerating up to a pre-defined speed, cruising at that speed during a pre-defined period of time and subsequently decelerating down to the original cruise speed. Any resolution trajectory involving a sequence of speed control actions acceleration-cruise-deceleration along the original flight path could be considered as fitting into the category represented by the conflict resolution pattern.

It is assumed that the conflicting aircraft can select only resolution trajectories that fit into the categories represented by the pre-defined conflict resolution patterns. The objective of grouping the possible resolution trajectories in patterns is to facilitate the prediction of the aircraft's responses to other aircraft's resolution trajectories. Besides fitting into one of the categories characterised by the conflict resolution patterns, the resolution trajectories produced by the trajectory-planning algorithm must be conflict-free with the resolution trajectories that have already been announced by other conflicting aircraft. These aircraft are referred to as *conflicting leaders*. Additionally, the algorithm operates under the assumption that its host aircraft will act as the leader of the remaining conflicting aircraft, which are referred to as *conflicting followers*. When searching for the appropriate resolution trajectory, the algorithm takes into account the conflicting followers' possible responses to the host's allowable resolution trajectories.

3.3.2.1 Selection of a *safe pattern*

The operation of the algorithm begins by assigning one of the pre-defined conflict resolution patterns to the host aircraft. Then, each conflicting follower is also assigned a pattern. Subsequently, the algorithm computes the inter-aircraft distances that would result from the host and the conflicting followers flying the patterns they have been assigned and the conflicting leaders flying their resolution trajectories. This computation is carried out under the assumption that the host aircraft acts as the leader and the conflicting followers respond in a specific sequence. The aim of computing the inter-aircraft distances is to ascertain whether the implementation of the assigned patterns would resolve the predicted conflicts.

The algorithm computes the inter-aircraft distances that result of the conflicting followers flying each possible combination of patterns and the conflicting leaders flying their resolution trajectories. These computations are carried out under the assumption that the host aircraft implements the pattern it was assigned initially and that the conflicting followers respond in the initially selected sequence. The algorithm returns the combinations of patterns that would result in the resolution of all the conflicts if they were flown by the conflicting followers. If the number of these combinations is greater than zero, the pattern initially assigned to the host aircraft is referred to as a *safe pattern* for the selected sequence.

Assuming that the conflicting followers always respond in the initially selected sequence, the computations described above are carried out assigning different patterns to the host aircraft. Once all the available patterns have been assigned to the host aircraft, the algorithm returns the conflict-free combinations of patterns for the conflicting followers corresponding to each safe pattern assigned to the host for the initially selected sequence.

Subsequently, a different sequence of conflicting followers responses is selected and the process above is carried out for that sequence. The algorithm returns the conflict-free combinations of patterns for the conflicting followers corresponding to each safe pattern assigned to the host for the new sequence. The algorithm conducts the entire process for each possible sequence of conflicting followers responses. Then, the algorithm selects a conflict resolution pattern that, if implemented by the host aircraft, would result in conflict-free patterns available for all the conflicting followers regardless of the sequence in which they respond. Thus, the selected pattern must be a safe pattern for all the possible sequences. It is required that the selected safe pattern allows for the highest possible number of conflict-free combinations of patterns for the conflicting followers assuming they respond following the *worst-case sequence of responses*. For a given safe pattern assigned to the host, the worst-case sequence of responses is the one that results in the minimum number of conflict-free combinations of patterns available to the conflicting followers.

3.3.2.2 Iterative improvement of the selected safe pattern

Once a safe pattern has been selected according to the criteria above, a cost function is defined. The cost function assigns a real-valued cost to each resolution trajectory according to operator's efficiency criteria such as flight time, fuel consumption and delay in arriving at a waypoint. Each aircraft-agent defines its own cost function according to its operator's specific preferences. The cost function provides a measure of the efficiency loss caused by the implementation of a resolution trajectory.

The objective in introducing a cost function is to guide the search for a resolution trajectory that contributes to the co-operative resolution of the conflict while causing the lowest possible efficiency loss. Unlike in Stackelberg games, the host aircraft's cost function is not influenced by the conflicting followers' reactions and hence the search for a Stackelberg equilibrium solution is not applicable. Instead, the algorithm searches for a trajectory that results in low cost for the host while facilitating the conflicting followers' search for their own low-cost responses. An *iterative improvement method* ([86], [87]) has been devised to enable the trajectory-planning algorithm to carry out the search. Iterative improvement methods are usually applied to optimisation problems in which finding a global optimal solution is either unnecessary or unattainable. When dealing with these particular problems, a solution that is sufficiently good according to certain criteria is generally acceptable. Iterative improvement methods provide a means to search for such a solution by improving an initial tentative solution through several iterative steps.

The iterative improvement method proposed here consists of repeatedly sampling at random from a subset of the allowable resolution trajectories during a pre-established time span. The subset considered comprises of the resolution trajectories that fit into the category represented by the selected safe pattern. The method operates as follows. First, the selected safe pattern itself is designated as the initial *candidate resolution trajectory*. Then, a resolution trajectory is selected at random from among those that fit into the category represented by the safe pattern. If this trajectory is conflict-free with the resolution trajectories of the conflicting leaders and allows for the same conflict-free combinations of patterns available to the conflicting followers as the selected safe pattern for all the possible sequences, then the trajectory is stored in memory. Otherwise the trajectory is discarded and a new trajectory of the type represented by the safe pattern is randomly sampled. This loop continues until a resolution trajectory fitting into the category of the safe pattern and meeting the above criterion is found and stored in memory. If such a trajectory has not been found after executing successively the above loop during the pre-established time span, then the search is halted and the selected safe pattern is proposed to the pilot as the resolution trajectory. If, on the other hand, a suitable trajectory is found and stored in memory, the value of the cost function for the trajectory is subsequently evaluated. If this cost is lower than the cost of the safe pattern, the trajectory stored in memory replaces it as the new candidate resolution trajectory. Otherwise the trajectory stored in memory is discarded.

Next, a new iteration begins with the generation of a new random resolution trajectory fitting into the category of the selected safe pattern. Successive iterations are performed until the pre-established time span expires. The candidate resolution trajectory in memory after the last iteration is proposed to the pilot as the resolution trajectory to be implemented.

The objective of the iterative improvement method described above is not to minimise the cost of the selected resolution trajectory. Instead, the output of the iterative process is a resolution trajectory that results in an acceptable cost for the aircraft while allowing for a wide range of potentially safe resolution trajectories among which the conflicting followers can search for their own low-cost resolution actions. The higher the number of iterations performed, the higher the probability of finding a candidate resolution trajectory with a lower cost, but also the longer the computation time and, consequently, the time lag between the conflict detection and the proposal of a resolution trajectory to the pilot. The time span during which the iterative process takes place represents a compromise between the aircraft-agent's drive to search for a low cost resolution trajectory and the need to provide the crew with an appropriate resolution action shortly after a conflict has been detected.

A flow chart describing schematically the proposed iterative improvement method is depicted in Figure 3.2.

3.3.2.3 Pseudocode for the trajectory-planning algorithm

The trajectory-planning algorithm is described in Figure 3.3 using *pseudocode* [88]. Pseudocode is a non-rigorous notation that resembles a programming language but that is not intended for actual computer compilation. It combines some of the logical structure of programming languages with a natural language to encode informal descriptions of the computations to be carried out. Pseudocode is particularly well suited to express algorithms as one need not to comply with the rigid syntax rules of a particular programming language and can concentrate on the definition of the sequence of instructions that constitute the algorithm. In addition, pseudocode is a valuable tool to support the implementation of an algorithm on a computer, since the translation of pseudocode into a programming language is usually straightforward.

3.4 Application of the proposed co-operation mechanism in two-dimensional conflicts

In this section, the co-operation mechanism described above will be applied in two-dimensional conflicts involving up to three Autonomous Aircraft cruising at the same altitude. The objective of applying the mechanism in such type of conflicts is to illustrate with simple examples its ability to support co-operative conflict resolution in the operational environment considered in this chapter.

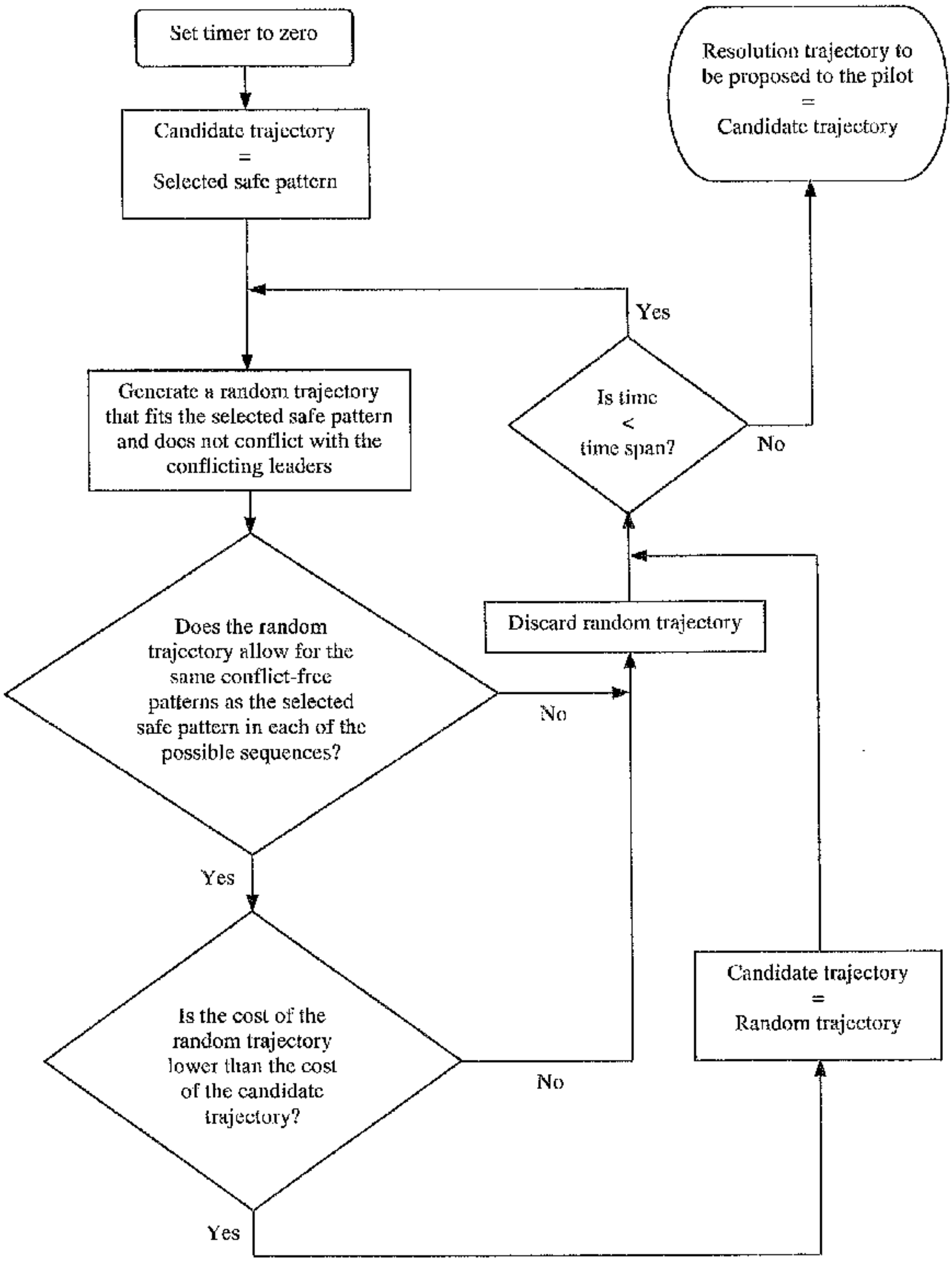


Figure 3.2: Flow chart of the iterative improvement process.

Procedure SelectSafePattern

Let n be the number of conflicting aircraft

Let A_0 represent the host aircraft

Let $A_1, A_2, \ldots, A_{n_c}$ represent the other $n_c = n-1$ aircraft involved in the conflict

Let $A_1, A_2, \ldots, A_{n_{cl}}$ represent the n_{cl} conflicting leaders

Let $T_{cl}^1, T_{cl}^2, \ldots, T_{cl}^{n_{cl}}$ represent the intended resolution trajectories of the conflicting leaders

Let $A_{n_{cl}+1}, A_{n_{cl}+2}, \ldots, A_{n_c}$ represent the n_{cf} conflicting followers, with $n_{cf} = n_c - n_{cl}$

Let $T_p^1, T_p^2, \ldots, T_p^{n_p}$ represent the n_p pre-defined conflict resolution patterns

Let $n_s = n_{cf}!$ be the number of possible sequences of action of the conflicting followers

Let $S_i = \langle A_{i_1}, A_{i_2}, \ldots, A_{i_{n_{cf}}} \rangle$ with $i \in \{1, 2, \ldots, n_s\}$ represent a possible sequence, where
  A_{i_1} reacts first, then A_{i_2} and so on

For each S_i **do**

  For each j_0 from 1 up to n_p **do**

   Assign $T_p^{j_0}$ to A_0

   $s_i^{j_0} = 0$

   For each j_1 from 1 up to n_p **do**

    Assign $T_p^{j_1}$ to A_{i_1}

    For each j_2 from 1 up to n_p **do**

     Assign $T_p^{j_2}$ to A_{i_2}

      $\ddots$

       For each $j_{n_{cf}}$ from 1 up to n_p **do**

        Assign $T_p^{j_{n_{cf}}}$ to $A_{i_{n_{cf}}}$

        For each pair $(k,l) \in \{0,1,\ldots,n_c\} \times \{0,1,\ldots,n_c\}$ do

         Let $d_{CPA}^{k,l}$ be the distance between A_k and A_l at their CPA

        If $d_{CPA}^{k,l} \geq$ Separation_Minimum for all pairs (k,l) **then do**

         $s_i^{j_0} = s_i^{j_0} + 1$

         Return $T_p^{j_0} \rightarrow$ Safe pattern for the host in sequence S_i

         Return $F_p(i, j_0, s_i^{j_0}) = \{T_p^{j_0}, T_{cl}^1, T_{cl}^2, \ldots, T_{cl}^{n_{cl}}, T_p^{j_1}, T_p^{j_2}, \ldots, T_p^{j_{n_{cf}}}\} \rightarrow$

         Set of conflict-free resolution patterns assuming that the conflicting
           followers respond to $T_p^{j_0}$ according to sequence S_i

Let T_{sp} be the set of conflict resolution patterns that are simultaneously safe patterns for
  the host in all the possible sequences

Search for a safepattern $T_p^r \in T_{sp}$ such that $\min_{i \in \{1,2,\ldots,n_s\}} s_i^r = \max_{T_p^m \in T_{sp}} \min_{i \in \{1,2,\ldots,n_s\}} s_i^m$

Procedure IterativeImprovement

Let $C_h(T_h)$ be the host's cost function, where T_h is a generic resolution trajectory

$T_h^r = T_p^r$

$C_h^r = C_h(T_p^r)$

t = Current_Time

ElapsedTime=0

While *ElapsedTime* $\leq$ Iteration_Time_Limit **do**

  Generate a random resolution trajectory T_h^{ran} that fits into the category represented by T_p^r

  if substituting T_h^{ran} for T_p^r in $F_p(i,r,s)$ results in a set of conflict-free resolution trajectories
    for all $s \in \{1, 2, \ldots, s_i^r\}$ and for all $i \in \{1, 2, \ldots, n_s\}$ **and** $C_h(T_h^{ran}) \leq C_h^r$ **then do**

   $T_h^r = T_h^{ran}$

   $C_h^r = C_h(T_h^{ran})$

  ElapsedTime = Current_Time-t

Output T_h^r and C_h^r

Figure 3.3: Pseudocode for the trajectory-planning algorithm.

In the two-dimensional conflict scenarios considered in this section, the Protection Volume is assumed to consist of a circle of radius 5 nautical miles centred on the aircraft, irrespectively of the look-ahead time for conflict detection. This 5 nautical miles separation minimum has been adopted inasmuch as it is currently used as a standard for radar separation. If the predicted distance between two aircraft at their nominal Closest Point of Approach is greater than 5 nautical miles, their intended trajectories are considered to be conflict-free. It is assumed that this separation minimum ensures that, despite deviations from the intended trajectories caused by guidance errors, the probability of a violation of the Collision Volume is greater than the Target Level of Safety.

In this section, no errors are assumed to occur in the conflict detection process and false alarms are not considered.

3.4.1 Synthesis of conflict resolution trajectories

The generic trajectory-planning algorithm has been implemented for two-dimensional conflict scenarios. The algorithm applies the point-mass equations of motion for a generic commercial aircraft [89] to generate potential conflict resolution trajectories. It is assumed that the only allowable conflict resolution actions are lateral manoeuvres, so that aircraft maintain their preferred airspeed and altitude while resolving conflicts. The use of speed changes during cruise to avoid conflict is considered excessively inefficient and has the potential of reducing the lifetime of the engines. To facilitate the resolution of the equations of motion, the mass of the aircraft is assumed to remain constant throughout the resolution process and the effect of the wind is not considered. Hence, the aircraft's ground speed is equal to its airspeed and remains constant during the resolution process.

Considering the above assumptions, the simplified point-mass equations of motion governing the generation of conflict resolution trajectories are as follows:

$$T = D \tag{3.3}$$

$$L = \frac{mg}{cos\,\phi} \tag{3.4}$$

$$\frac{dm}{dt} = 0 \tag{3.5}$$

$$\frac{d\Psi}{dt} = \frac{L \sin\phi}{mV_g} \tag{3.6}$$

$$\frac{dx}{dt} = V_g \sin\Psi, \quad \frac{dy}{dt} = V_g \cos\Psi, \quad \frac{dz}{dt} = 0 \tag{3.7}$$

where T is the engine thrust, D is the drag, L is the lift, m is the aircraft mass, g is the acceleration of gravity, Ψ is the aircraft heading measured clockwise from North, ϕ is the bank angle, V_g is the aircraft's ground speed, x and y are the aircraft's position along two earth-fixed orthogonal axes aligned with the geographical south-north axis and the geographical west-east axis, respectively, and z is the aircraft's altitude.

The algorithm models the heading changes contained in the resolution trajectories as *inside turns* [89]. Inside turns emulate heading changes calculated by Flight Management Systems. Using inside turns in the generation of resolution trajectories results in manoeuvres that are compatible with the FMS and the pilots are familiar with. An inside turn can be modelled by a circular arc tangent to the lines connecting three waypoints, as it is shown in Figure 3.4.

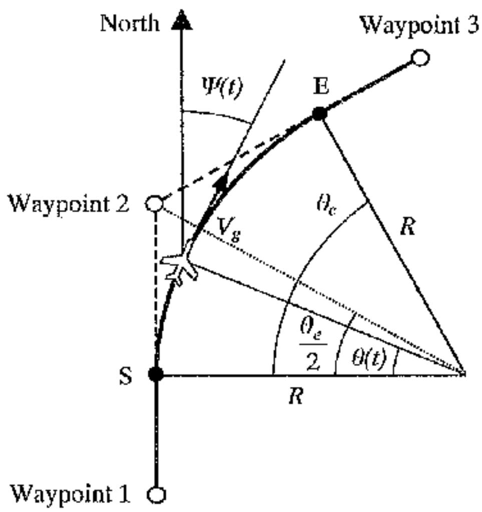


Figure 3.4: Generic inside turn used in the trajectory generation process.

To generate an inside turn, the trajectory-planning algorithm integrates the equations (3.7) considering the function $\Psi(t)$ obtained by integrating the following expression:

$$\frac{d\Psi}{dt} = \frac{d\theta}{dt} = \frac{V_g}{R} \qquad (3.8)$$

where R is the radius of the inside turn (see Figure 3.2). R can be calculated substituting equation (3.8) into equation (3.6), which results in:

$$R = \frac{V_g^2}{g\,tan\phi} \qquad (4.9)$$

Considering the standard operational performance of civil aircraft in level turns during cruise, it is assumed that the bank angle ϕ remains constant and equal to 25° during an inside turn [90]. For simplicity, it is also assumed that the change in bank angle at the start of the turn is instantaneous. Thus, the integration of equation (3.8) results in:

$$\Psi(t) = \left(\frac{g}{V_g} tan\phi_t\right)t + \Psi_S \qquad (4.10)$$

where Ψ_S is the aircraft heading at the start of the turn and $\phi_t = 25^{\circ}$. The co-ordinates of the points S and E in Figure 3.4, which correspond to the start and the end of the turn, respectively, are determined considering the value of R from (3.9) and assuming that Waypoint 2 marks the midpoint of the turn. As shown in Figure 3.2, the value of $\theta(t)$ corresponding to Waypoint 2 is

$$\theta(t)_{\text{Waypoint 2}} = \frac{\theta_e}{2} \qquad (3.11)$$

The angle θ_e is given by

$$\theta_e = \Psi_E - \Psi_S \qquad (3.12)$$

where Ψ_E is the aircraft heading at the end of the turn, which is equal to the aircraft heading at Waypoint 3 and Ψ_S is the aircraft heading at the start of the turn, which is equal to the aircraft heading at Waypoint 1.

3.4.1.1 Allowable conflict resolution trajectories

In principle, if the equations and assumptions introduced above are applied to generate conflict resolution trajectories, then any sequence of straight lines and circular arcs governed by those equations and complying with those assumptions can be considered as a potential resolution trajectory. To facilitate the operation of the trajectory-planning algorithm, only a subset of all those potential resolution trajectories is considered. In the conflict scenarios considered in this chapter, it is assumed that the only resolution trajectories generated by the trajectory-planning algorithm are *lateral shift manoeuvres* constructed according to the equations and assumptions above. A lateral shift manoeuvre consists of a lateral deviation from the preferred straight path followed by the resumption of the initially intended route. Such a manoeuvre involves three heading changes. The trajectory-planning algorithm models lateral shift manoeuvres as sequences of straight lines connected by inside turns. Figure 3.5 shows a generic lateral shift manoeuvre as modelled by the algorithm.

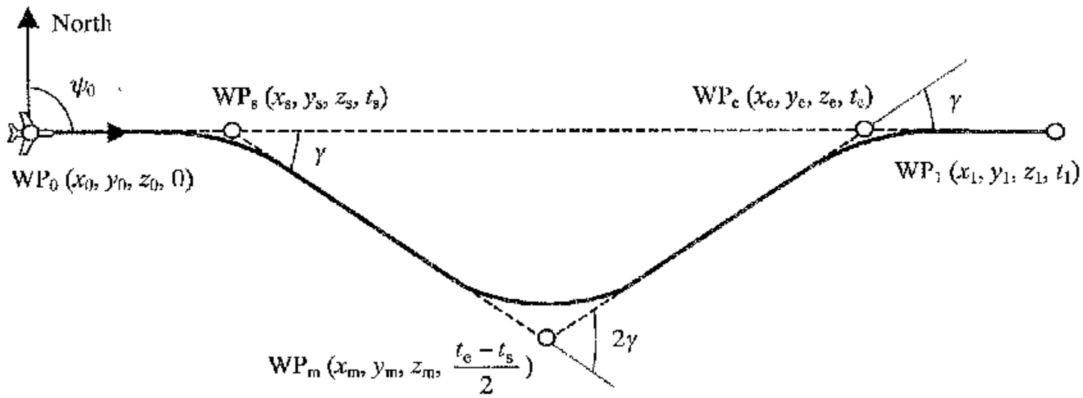


Figure 3.5: Generic lateral shift manoeuvre.

A generic lateral shift manoeuvre is determined by the waypoints WP_0, WP_s, WP_m WP_e, and WP_1, as depicted in Figure 3.5. WP_0 and WP_1 mark the start and the end of the resolution manoeuvre, respectively, while WP_s, WP_m and WP_e mark the midpoints of the three inside turns that the manoeuvre involves. WP_0 corresponds to the time when the aircraft initiates the planning process. The times at the different waypoints are measured from the time at WP_0. It is assumed that the distance between WP_s and WP_m equals the distance between WP_m and WP_e. Thus, given the aircraft's initial four-dimensional position, WP_0, and its initial heading, ψ_0, the only parameters needed to generate a lateral shift manoeuvre according to Figure 3.5 are the angle γ and the times at the waypoints WP_s and WP_e, namely t_s and t_e.

The value of the angle γ defines the magnitudes of the heading changes that the lateral shift manoeuvre involves. It is assumed that positive values of the angle γ correspond to right lateral shift manoeuvres while negative values of the angle γ correspond to left lateral shift manoeuvres. Right lateral shift manoeuvres cause the aircraft to deviate towards the right with respect to its originally intended course while left lateral shift manoeuvres cause the aircraft to deviate towards the left with respect to its initially intended route. The time t_s marks the midpoint of the first inside turn of the lateral shift manoeuvre and indirectly determines when the turn is initiated. The time t_e marks the midpoint of the last inside turn of the lateral shift manoeuvre and indirectly determines the time at which the aircraft returns to its original course. Both times are measured with respect to the time at WP_0.

Circumscribing the potential resolution actions to lateral shift manoeuvres increases the predictability of the conflicting followers' reactions to the resolution trajectories of their conflicting leaders. This facilitates the search for resolution trajectories expected to contribute to the co-operative resolution of the conflicts. An additional advantage of limiting the potential resolution actions to lateral shift manoeuvres is that the resulting resolution trajectories are only temporary deviations from the aircraft's original flight plans and do not involve major changes to their preferred routes.

3.4.2 Model of the flight crew response latency

As soon as an aircraft detects a conflict, its flight crew is made aware of it through an appropriate alerting procedure, which may include visual information, cues depicted on the CDTI and aural annunciations. Shortly thereafter, the flight crew is proposed a resolution action that is expected to contribute to the co-operative resolution of the predicted conflict. This resolution action consists of a modification of the currently intended trajectory in the form of a lateral shift manoeuvre. The information needed for the flight crew to understand the proposed trajectory is depicted in the CDTI. This information includes the positions of the waypoints that define the resolution trajectory, namely WP_0, WP_s, WP_m WP_c, and WP_1. In nominal operation, the flight crew first becomes aware of the detected conflict, then comprehends the proposed resolution manoeuvre and its implications, and finally accepts to modify the aircraft's intended route according to the proposed resolution trajectory. As soon as the flight crew accepts the proposed resolution trajectory, a flag indicating a change in the aircraft intent together with the new intent are broadcast via ADS-B.

The resolution trajectories proposed to the flight crew must incorporate a time buffer to accommodate the *flight crew response latency,* which refers to the time elapsed between the detection of a conflict and the acceptance of the proposed resolution trajectory. A model of the flight crew response latency is needed to ensure that, in nominal operation, the proposed resolution trajectory allows sufficient time for the crew to accept it prior to the time at which the aircraft is scheduled to start deviating from its originally intended route. As it has been indicated previously, this time is determined by the time at the waypoint WP_s, t_s, which is one of the three parameters needed to define a resolution trajectory. Once a model of the flight crew response latency has been adopted, a minimum allowable value for t_s will be defined on the basis of the model.

The work described in [54] proposes a probabilistic model of the flight crew response latency for airborne separation assurance in a Free Flight environment. The time elapsed between the issue of a conflict alert and the initiation of a resolution manoeuvre is modelled as a gamma distribution with a mean of 60 seconds and a variance such that there is a 95% probability that the response occurs within 120 seconds. This relatively long latency is deliberately designed to allow time to co-ordinate with other aircraft and/or ATC. Based on this work, a simple probabilistic model of the flight crew latency has been adopted for the conflict scenarios considered in this chapter. The time elapsed between the detection of a conflict and the acceptance of the proposed resolution trajectory by the flight crew in nominal operation is modelled by an uniform distribution with the probability density function shown in Figure 3.6. According to this distribution, the value of the response latency can take any value between 40 and 80 seconds with equal likelihood. This results in the flight crew accepting the proposed resolution trajectory within 80 seconds of the conflict detection with a probability of 100%. The upper bound for the response latency in the model adopted is lower than the

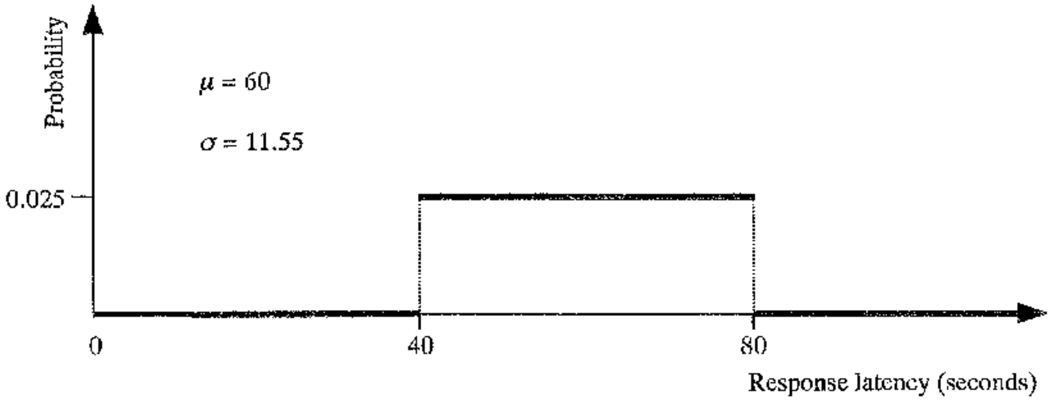


Figure 3.6: Probability density function of the flight crew response latency.

one in [54]. This is justified by the fact that the conflict resolution methodology proposed here does not involve explicit co-ordination either between flight crews or between the flight crew and ATC. According to the model adopted, the value of t_s must be sufficiently greater than 80 seconds so that the first turn of the resolution trajectory is not scheduled to start within the first 80 seconds of the conflict detection. This ensures that the flight crew has enough time to become aware of the detected conflict, understand the proposed resolution trajectory and accept it before the aircraft is due to start turning.

In the conflict scenarios considered here, the value chosen for the minimum allowable t_s is 120 seconds. This value is intended to provide an additional time lag between the acceptance of the proposed resolution manoeuvre and the time at which the aircraft is scheduled to start deviating from its original course. This additional time lag could allow for a re-planning of the resolution trajectory in the event of that trajectory giving rise to a conflict with an aircraft detected after the trajectory has been accepted by the crew and before the first turn is initiated. In such an event, the accepted resolution trajectory would be cancelled and a new one proposed to the flight crew considering the intended trajectory of the new aircraft. In addition, setting the minimum allowable value for t_s to 120 seconds ensures that the flight crew is not presented with a resolution trajectory that involves starting a turn excessively soon after the acceptance of that trajectory. This would contribute to reduce the urgency to resolve conflicts that could be experienced by the flight crew. Since conflict resolution will be a matter of course for flight crews in future AAO, the perceived urgency related to conflict alerts should be minimised to avoid undesirable levels of stress in the cockpit when performing separation assurance tasks. Similarly to the current ATC-based separation assurance, ASAS-based conflict resolution in AAO should be regarded as a fairly routine operation rather than as an emergency procedure.

For simplicity, it is assumed that the model of the flight crew response latency in Figure 3.6 also applies to the time elapsed between a conflicting follower's reception of a conflicting leader's resolution trajectory and the acceptance of the response to the received trajectory by the conflicting follower's crew.

3.4.3 Conflict resolution patterns

In the conflict scenarios considered in this chapter, lateral shift manoeuvres are the only allowable actions to resolve conflicts. The aircraft-agents know that their conflicting aircraft-agents generate only lateral shift manoeuvres as candidate resolution trajectories. As explained in section 3.3, the trajectory-planning algorithm relies on a set of conflict resolution patterns known to all the aircraft-agents to facilitate the prediction of the conflicting aircraft's possible responses to the chosen resolution trajectory. A conflict resolution pattern is a trajectory considered representative of all the resolution trajectories that can be classified as belonging to the same category. The allowable resolution trajectories are categorised into three different types and each of these types is represented by a conflict resolution pattern.

The three patterns are schematically depicted in Figure 3.7. Pattern 1, which coincides with the initially intended trajectory, is a lateral shift manoeuvre with $\gamma = 0^{\circ}$. This pattern represents the situations in which no action is taken in response to a conflict. Patterns 2 and 3 represent left and right lateral shift manoeuvres, respectively. These two patterns are two symmetrical lateral shift manoeuvres determined by the values of the parameters γ^{p}, t_{s}^{p} and t_{e}^{p}, which are known to all the aircraft-agents.

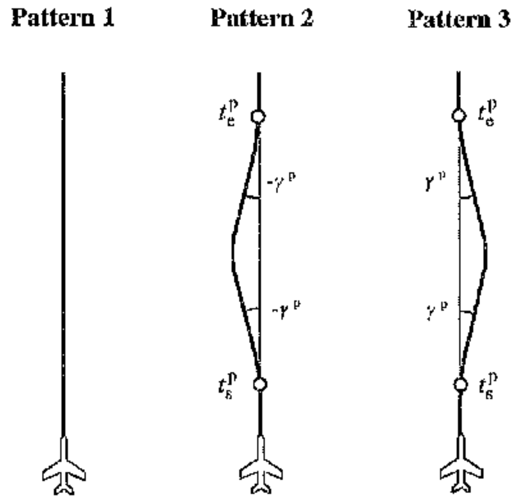


Figure 3.7. Conflict resolution patterns

The value of γ^{P} determines the magnitudes of the heading changes that the patterns involve. It is assumed that $\gamma^{\mathrm{P}}>0$. Initially, γ^{P} is set to 10°. If this value results in no safe patterns, γ^{P} is successively incremented by 5° until at least one safe pattern is identified. The value of t_{s}^{P} is the time at the waypoint $\mathrm{WP_s}$ for both patterns as measured from the time at which the aircraft initiates the planning process. This time is set to 120 seconds. The time t_{e}^{P} is the time at the waypoint $\mathrm{WP_e}$ for both patterns 2 and 3 as measured from the time when the aircraft initiates the planning process. The value of t_{e}^{P} is defined as a function of the average of the times of the Closest Points of Approach in all the conflicts in which the aircraft is involved. This average time is denoted by t_{CPA}^{av}. It is assumed that the time at the waypoint $\mathrm{WP_m}$ for both patterns must coincide with t_{CPA}^{av}. Therefore, t_{e}^{P} can be expressed as

$$t_{e}^{\mathrm{P}} = t_{\mathrm{CPA}}^{\mathrm{av}} + \left(t_{\mathrm{CPA}}^{\mathrm{av}} - t_{s}^{\mathrm{P}}\right) \tag{3.13}$$

3.4.4 Cost function

The resolution trajectory produced by the trajectory-planning algorithm is the result of an iterative improvement process based upon a cost function that assigns a real-valued cost to each allowable resolution trajectory. The cost of a resolution trajectory depends on the operational criteria of the airline operating the aircraft. For example, in a particular situation an airline might prefer an increase in fuel consumption to an increase in flight time, since the latter could cause missing its landing slot at the arrival airport. Thus, each aircraft may apply a different cost function according to its operator's preferences. In the scenarios considered in this chapter, it is be assumed that all the conflicting aircraft use the same type of cost function. The reason for this assumption is twofold. Firstly, to establish a reference for the comparative analysis of different resolution strategies provided by trajectory-planning algorithm. Secondly, to illustrate quantitatively the capability of the co-operation mechanism to support the sharing of the costs of the conflict resolution. The following function is proposed as the single cost function to be applied by all the conflicting aircraft:

$$C(\gamma,t_s,t_e) = w_{\mathrm{D}}D(\gamma,t_s,t_e) + w_{\mathrm{M}}M(\gamma) \tag{3.14}$$

The function $D(\gamma, t_s, t_e)$ relates to the losses resulting from the deviation from the user-preferred route caused by the implementation of a resolution manoeuvre. The performance of a lateral shift manoeuvre involves flying over a longer distance than originally intended. This causes an increase in the amount of fuel burnt and may induce delays at future waypoints and even at the arrival airport. The function $D(\gamma, t_s, t_e)$ provides a non-dimensional measure of the deviation from the user-preferred route induced by a given lateral shift manoeuvre rather than an account of the economic losses caused by the deviation. Thus, the function $D(\gamma, t_s, t_e)$ is defined as a non-dimensional quantity proportional to the distance between the aircraft's predicted position at the time of completion of the lateral shift manoeuvre and the aircraft's predicted position at the same time if it continued along its preferred route. Graphically, this distance, which is denoted as d_c, corresponds to the distance between RT(t_1) and UT(t_1) in Figure 3.8. Therefore, $D(\gamma, t_s, t_e)$ is given by

$$D(\gamma,t_s,t_e) = K_{\mathrm{D}}d_c, \quad K_{\mathrm{D}} > 0 \tag{3.15}$$

where K_{D} is a constant of proportionality.

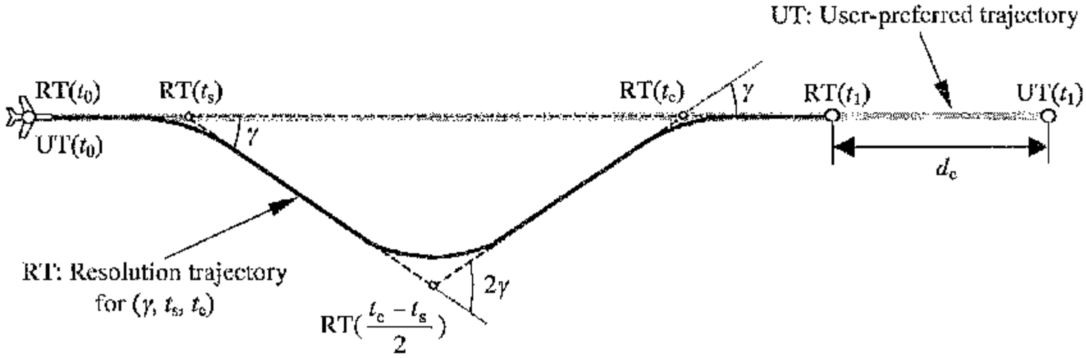


Figure 3.8. Definition of the distance d_e.

The function $M(\gamma)$ has been added to the cost function to represent possible losses derived directly from the heading changes that the resolution trajectories involve. These losses refer to the discomfort that lengthy turns can cause to passengers and to the possibility of new conflicts arising as a result of the excessive lateral deviation from the original course caused by resolution trajectories involving pronounced heading changes. Since the value of the angle γ provides a measure of the heading changes that the execution of a lateral shift manoeuvre involves, the function $M(\gamma)$ has been defined as a non-dimensional quantity proportional to the absolute value of the angle γ:

$$M(\gamma) = K_M|\gamma|, \quad K_M > 0 \tag{3.16}$$

where K_M is a constant of proportionality.

The factors w_D and w_M in the cost function (3.14) represent the weights the aircraft-agent assigns to $D(\gamma, t_s, t_e)$ and $M(\gamma)$, respectively. These weights indicate the relative importance attributed to each of the functions. The values of w_D and w_M are subject to the following constraints:

$$w_D \in [0,2], \; w_M \in [0,2], \; w_D + w_M = 2 \tag{3.17}$$

In principle, it is assumed that all the conflicting aircraft assign equal weights to both functions. Considering the constraints in (3.17), this assumption results in $w_D = w_M = 1$. However, an exception to this assumption will be made in one example below, in which each of the two conflicting aircraft assigns a different pair of weights to the functions $D(\gamma, t_s, t_e)$ and $M(\gamma)$, and therefore they do not apply the same cost function. This example is intended to illustrate that the co-operation mechanism is applicable independently of the cost functions used by the conflicting aircraft.

3.4.5 Two-dimensional conflicts between two aircraft

Considering the foregoing, this section illustrates the application of the proposed co-operation mechanism in two-dimensional conflicts between two aircraft. The performance of the mechanism in this type of conflict scenario is analysed for different conflicting configurations. The analysis focuses on the probabilistic aspects of the mechanism as well as on its capacity to enable the conflicting aircraft to share the resolution costs.

3.4.5.1 Generic conflict scenario

The generic conflict scenario considered for the analysis of the application of the co-operation mechanism in two-dimensional conflicts involving two aircraft is depicted in Figure 3.9. The scenario involves two aircraft A_1 and A_2 flying at constant speeds at the same altitude along straight intersecting paths. The path-crossing angle is denoted by α. The positions of the aircraft are referred to two earth-fixed orthogonal axes X and Y aligned with the geographical south-north axis and the geographical west-east axis, respectively. The aircraft's initial heading and airspeed determine their intended routes. In Figure 3.9, H_1 and V_1 denote A_1's heading and airspeed, respectively, and H_2 and V_2 denote A_2's heading and airspeed, respectively.

Once they are within each other's ADS-B range of coverage, the aircraft compare their intended trajectories and ascertain that the anticipated nominal inter-aircraft distance at the CPA, d_{CPA}, is smaller than 5 nautical miles. For simplicity, it is assumed that both aircraft have the same ADS-B range of coverage and the conflict is detected simultaneously on board both aircraft. The initial inter-aircraft distance, d_d, corresponds to the time at which the conflict is detected by both aircraft. It will be shown below that small differences in the times at which each of the aircraft detects the conflict do not

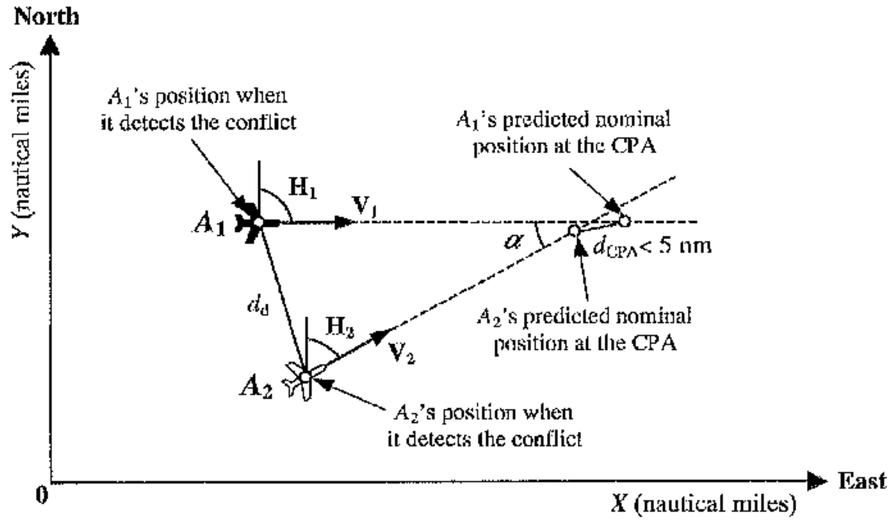


Figure 3.9. Generic conflict scenario involving two aircraft

affect the overall performance of the co-operation mechanism. Non-nominal scenarios involving one of the conflicting aircraft operating with a degraded ADS-B range of coverage might require additional operational procedures not included in the co-operation mechanism. Such non-nominal scenarios are not considered in this thesis and might be the subject of future research.

3.4.5.2 The trajectory-planning algorithm in the generic conflict scenario considered

A version of the trajectory-planning algorithm specifically adapted to the generic conflict scenario introduced above has been implemented. The considerations and assumptions regarding the available resolution actions, the flight crew response latency, the conflict resolution patterns and the cost function discussed in the previous sections have been incorporated into the algorithm. In addition, the process of selecting the conflict resolution pattern that serves as the input to the iterative improvement process has been simplified.

In the generic conflict scenario considered here, there is only one conflicting follower and therefore only one possible sequence of responses. This circumstance can be taken into account to simplify the selection of a safe pattern. Firstly, the host attempts to select a pattern that is conflict-free when pattern 1 is assigned to the conflicting follower. If none of the three pre-defined patterns satisfies this condition, then the algorithm applies the criterion of the generic algorithm described in section 3.3 and searches for a pattern that is conflict-free with the maximum possible number of patterns assigned to the conflicting follower.

The safe pattern selection process described above allows for some conflicts to be resolved with resolution strategies in which one of the two aircraft flies a low-cost resolution trajectory and the other aircraft maintains its initially intended route. In spite of the fact that one aircraft bears the total resolution cost, such resolution strategies can still be considered co-operative since they enable the aircraft to co-ordinately contribute to the achievement of satisfactory global performance. Besides, the aircraft that flies the resolution trajectory can be seen as acting helpfully towards the aircraft that does not modify its initially intended route. An additional advantage of this type of resolution strategies is that the conflicting aircraft that does not have to modify its initially intended route is readily available to perform an emergency resolution action, should it be necessary in non-nominal situations. Such non-nominal situations may arise as a consequence of a malfunction preventing the manoeuvring aircraft from performing its resolution action accurately or as a result of a conflict with a third conflicting aircraft being detected after the resolution trajectory has been established.

Once a safe pattern has been selected, the iterative improvement process is initiated. The process requires the generation of random resolution trajectories that fit the selected pattern. Inasmuch as the three parameters γ, t_s and t_e univocally define a resolution trajectory, generating a random resolution trajectory is equivalent to assigning random values to these three parameters. The random values for a parameter are sampled from the set of allowable values for that parameter. Since the generated trajectories must belong in the category defined by the selected safe pattern, the set of the allowable values for a parameter depends on that pattern. For simplicity, the sets of allowable values for the parameters are discrete. The set of allowable values for the parameter γ is denoted as Γ and is defined as follows:

$$
\left.\begin{array}{ll}
\text{If pattern 1 is the selected safe pattern:} & \Gamma=\{0\} \\
\text{If pattern 2 is the selected safe pattern:} & \Gamma=\{-\gamma^{p}, -\gamma^{p}+1, \ldots, -6, -5\} \\
\text{If pattern 3 is the selected safe pattern:} & \Gamma=\{5, 6, \ldots, \gamma^{p}-1, \gamma^{p}\}
\end{array}\right\} \quad (3.18)
$$

where the elements of Γ are expressed in degrees. As it can be seen in (3.18), it is assumed that $|\gamma| \geq 5^{o}$ for patterns 2 and 3.

Considering that, in a conflict involving two aircraft, the time t_{CPA}^{av} coincides with the time of the Closest Point of Approach between the two aircraft, the set of allowable values for t_s is defined as follows:

$$
\mathrm{T_s}=\{t_s^p, t_s^p+1, t_s^p+2, \ldots, t_{\mathrm{CPA}}-2, t_{\mathrm{CPA}}-1, t_{\mathrm{CPA}}\} \quad (3.19)
$$

where t_{CPA} denotes the time of the Closest Point of Approach measured from the time when the aircraft initiates the planning process. The elements of $\mathrm{T_s}$ are expressed in seconds. The set of allowable values for t_e is defined as follows:

$$
\mathrm{T_e}=\{t_{\mathrm{CPA}}, t_{\mathrm{CPA}}+1, t_{\mathrm{CPA}}+2, \ldots, t_e^p-2, t_e^p-1, t_e^p\} \quad (3.20)
$$

where the elements of $\mathrm{T_e}$ are expressed in seconds. Let γ^r, t_s^r and t_e^r be three values randomly sampled from Γ, $\mathrm{T_s}$, and $\mathrm{T_e}$, respectively. It is assumed that these values represent a valid resolution trajectory only if

$$
t_e^r - t_s^r \geq 120 \text{ s} \quad (3.21)
$$

This condition has been imposed to allow for sufficient time between the first and the last turn so that the generated resolution trajectory is feasible.

3.4.5.3 Simulation of the application of the co-operation mechanism

The version of the trajectory-planning algorithm described above has been implemented using the MATLAB® computing language [91] to simulate the resolution of conflicts according to the proposed co-operation mechanism in the scenario considered. The MATLAB® integrated computing environment facilitates the coding of the algorithm and the visualisation of resolution trajectories.

To simulate the process of resolving a conflict, it is first necessary to specify the response latencies of the flight crews, which determine which of the two aircraft will act as the leader and which one will act as the follower. The simulation begins by running the trajectory-planning algorithm for each of the two conflicting aircraft considering the specified response latencies of the flight crews. Subsequently, the resulting resolution trajectories are generated and visualised. To analyse the performance of the mechanism in a given scenario, the application of the co-operation mechanism is simulated for different flight crew response latencies.

The simulations are performed using the version 5.2 of MATLAB® for Windows® 95 on a desktop PC with 64 MB RAM equipped with an Intel® Pentium® MMX processor operating at 233 MHz CPU clockspeed. The CPU time limit for the iterative improvement process has been set to 10 seconds. As a result, the algorithm takes between 12 and 22 seconds to produce a resolution trajectory. Such a time lag between the conflict detection and the proposal of a resolution action is regarded as acceptable considering the model of the flight crew response latency that has been adopted. In principle, a longer iteration time would result in resolution trajectories with a lower cost but would also mean longer delays in the proposal of the trajectories to the flight crew. With the increasing computing power available onboard modern aircraft, it is anticipated that the time limit for the iterative improvement process could be reduced in an airborne implementation of the algorithm.

3.4.5.4 Application of the co-operation mechanism in conflict scenario 1

This section illustrates how the co-operation mechanism enables the two aircraft in conflict scenario 1, depicted in Figure 3.10, to co-ordinate their resolution trajectories and share the total resolution costs. To do so, the application of the mechanism in this conflict scenario has been simulated as described above.

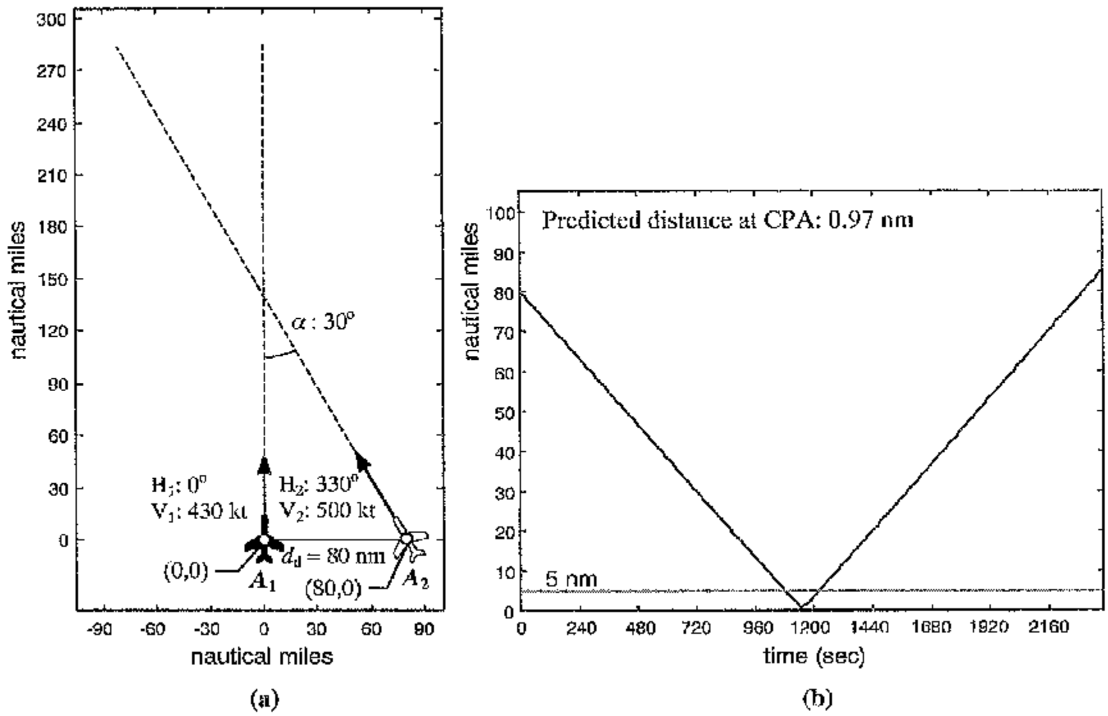


Figure 3.10. Conflict scenario 1. (a) Initial configuration (b) Predicted distances between the aircraft as they fly along their initially intended routes

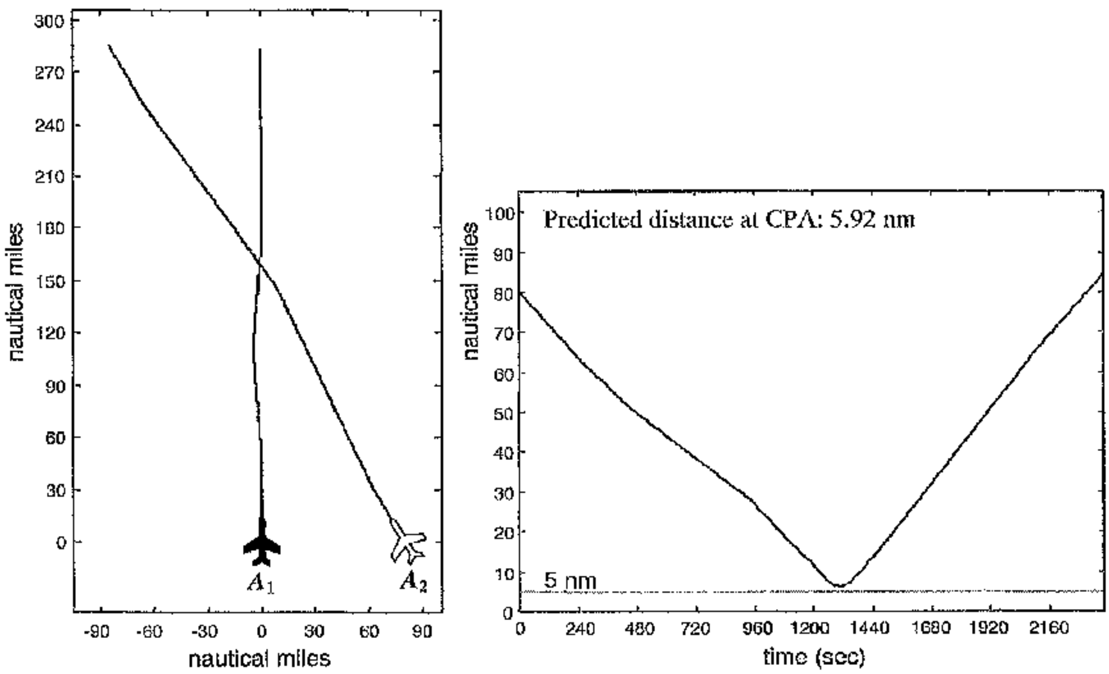
In the simulation, it is assumed that both aircraft-agents A_1 and A_2 apply the same cost function

$$C(\gamma, t_s, t_c) = D(\gamma, t_s, t_c) + M(\gamma) \tag{3.22}$$

The cost function (3.22) corresponds to assigning equal weights to both $D(\gamma, t_s, t_c)$ and $M(\gamma)$, and results from substituting $w_D = w_M = 1$ into the generic cost function (3.14). It is also assumed that A_1 acts as the leader and its flight crew accepts the proposed resolution trajectory 60 seconds after conflict detection, which is consistent with the model of the flight crew response latency adopted. When A_2 receives A_1's resolution trajectory, it regards itself as the follower of A_1, informs its flight crew of the new situation and starts planning a resolution action that is conflict-free with A_1's new intentions.

It is implicitly assumed that A_1 detects the conflict at the time of the initial configuration depicted in Figure 3.10(a). The exact time at which A_2 detects the conflict as well as the time it takes for its flight crew to assess and accept its response to A_1's resolution action are irrelevant to the simulation results. Since A_1 is assumed to act as the leader, A_2 elaborates its resolution trajectory after receiving A_1's resolution trajectory. Thus, A_2's resolution trajectory does not depend on the time when A_2 detects its conflict with A_1. In nominal operation, the flight crew of A_2 accepts the resolution trajectory no more than 80 seconds after receiving A_1's resolution trajectory. Since the trajectory-planning algorithm produces resolution trajectories that do not require A_2 to start deviating from its initial route until at least 120 seconds after receiving A_1's resolution trajectory, the flight crew of A_2 is guaranteed to accept the resolution trajectory before the aircraft is due to start manoeuvring.

The resolution trajectories that result from the simulation together with the predicted distances between the aircraft as they fly those trajectories are shown in Figure 3.11. Table 3.2 displays the parameters defining the two resolution trajectories as well as the costs associated to each of them and their sum, which can be considered as the total cost of resolving the conflict.



(a) (b)

Figure 3.11. Simulation of the resolution of the conflict in scenario 1: A_1 acts as the leader and its flight crew accepts the proposed resolution trajectory 60 s after the detection of the conflict. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

	γ	t_s	t_e	**Cost**
A_1	-5°	447 s	1424 s	1.69
A_2	5°	248 s	2114 s	2.24
Total conflict resolution cost				3.93

Table 3.2: Values of parameters and costs of the resolution trajectories in Figure 3.11(a)

As shown in Table 3.2, the co-operation mechanism enables A_1 and A_2 to share the total cost of resolving the conflict. Both A_1 and A_2 contribute to the resolution of the conflict and assume part of the total cost. To further illustrate this feature, the simulation of the application of the co-operation mechanism has been run for the same scenario imposing that A_2 maintains its initially intended trajectory. A_1 applies the trajectory-planning algorithm to search for a resolution trajectory that solves the conflict without A_2 needing to act. Again, the simulation assumes that A_1 acts as the leader and its flight crew accepts the proposed resolution trajectory 60 seconds after conflict detection. The results of the simulation are shown in Figure 3.12. Table 3.3 displays the parameters defining the resolution trajectories in Figure 3.12(a) as well as the costs associated to each of them and their sum. In this case, A_1 resolves the conflict without any contribution from A_2, thereby assuming all costs associated with the conflict resolution. The cost of A_1's resolution trajectory, which is the total cost of the resolution, is higher than the total cost shared by A_1 and A_2 in the conflict resolution strategy depicted in Figure 3.11.

The resolution of the conflict in scenario 3.1 has been simulated assuming that A_1 and A_2 assign different values to the weights in the cost function in (3.14) to illustrate that the co-operation mechanism is applicable in scenarios in which the conflicting aircraft apply different cost functions. In this case, the weights in A_1's cost function are $w_D = w_M = 1$, while those in A_2's cost function are $w_D = 0$ and $w_M = 2$. As in the previous examples, the simulation assumes that A_1 acts as the leader and its flight crew accepts the proposed resolution trajectory 60 seconds after conflict detection. The results of the simulation are shown in Figure 3.13. Table 3.4 shows the parameters defining the resolution trajectories in Figure 3.13(a) as well as their associated costs.

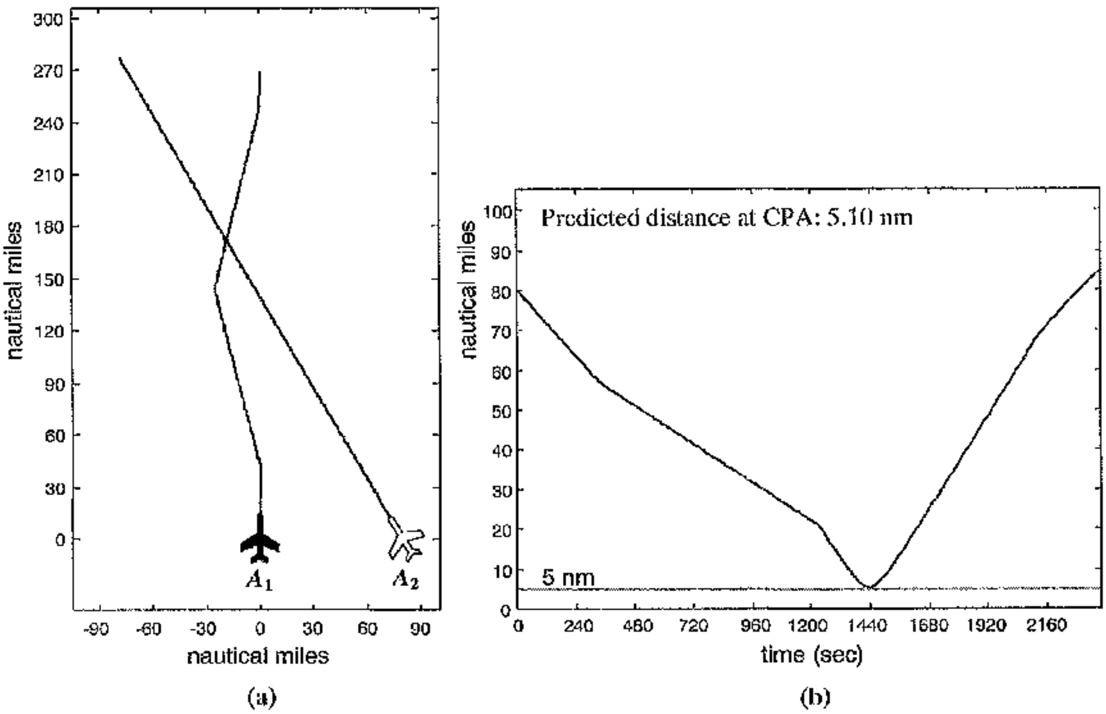


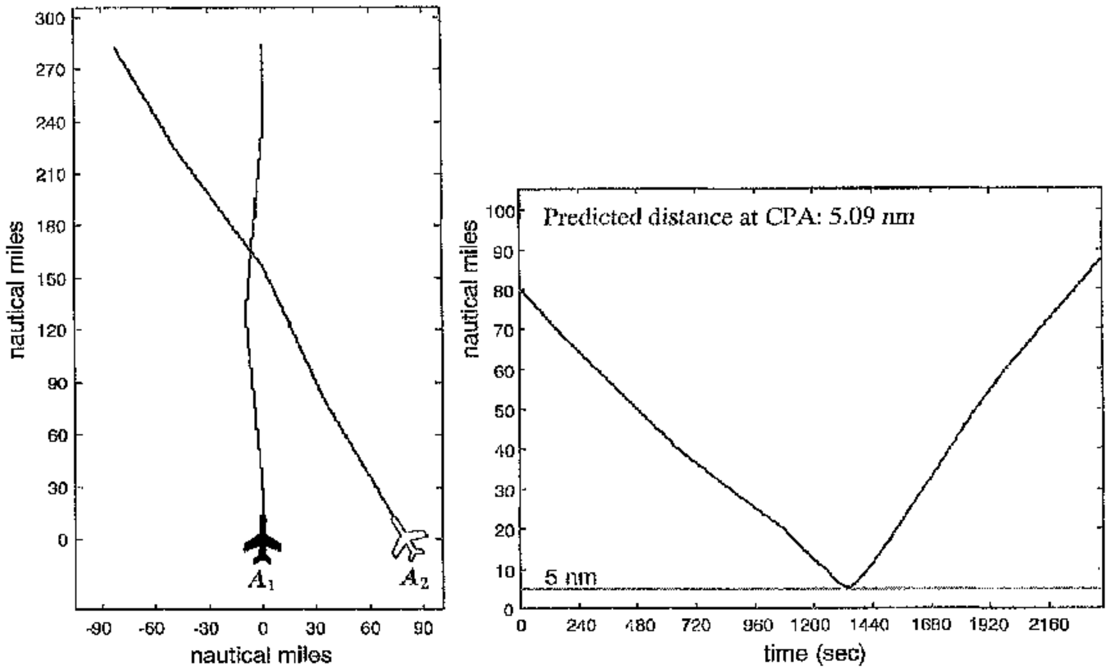
Figure 3.12. Simulation of non-co-operative resolution of the conflict in scenario 1: A_1 plans a resolution trajectory that does not require A_2 to act; A_1's flight crew accepts the proposed resolution trajectory 60 s after the detection of the conflict. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

	γ	t_s	t_e	**Cost**
A_1	-14°	334 s	2128 s	9.86
A_2	Initially intended route			0.00
Total conflict resolution cost				9.86

Table 3.3: Values of parameters and costs of the resolution trajectories in Figure 3.12(a)

With the objective of establishing a common reference for comparing the cost of applying the co-operation mechanism in different situations, in the remainder of this chapter it will be assumed that all the conflicting aircraft in all the conflict scenarios considered apply the cost function (3.22).

Figure 3.14 shows the result of simulating the application of the co-operation mechanism in scenario 1 assuming that A_2 acts as the leader and its flight crew accepts the proposed resolution trajectory 60 seconds after it detects the conflict. In this case, it



(a) (b)

Figure 3.13. Simulation of the resolution of the conflict in scenario 1: A_1 acts as the leader and its flight crew accepts the proposed resolution trajectory 60 s after the detection of the conflict; the cost function applied by A_2 is different from the one applied by A_1. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

	γ	t_s	t_e	Cost
A_1	-5°	155 s	2022 s	2.10 ($w_D = 1$, $w_M = 1$)
A_2	6°	660 s	1886 s	1.87 ($w_D = 0$, $w_M = 2$)
Total conflict resolution cost				3.97

Table 3.4: Values of parameters and costs of the resolution trajectories in Figure 3.13(a).

is implicitly assumed that A_2 detects the conflict at the time of the initial configuration depicted in Figure 3.10(a). Table 3.5 shows the parameters defining the resolution trajectories in Figure 3.14(a) as well as their associated costs. In this case, the conflict is resolved without A_1 having to react to A_2's resolution action. With A_2 acting as the leader, the resolution strategy results in a slightly higher cost for A_2 in comparison to the strategy obtained with A_1 acting as the leader (see Figure 3.11 and Table 3.2). However, A_1 does not have to modify its preferred route and the total cost of resolving the conflict is lower.

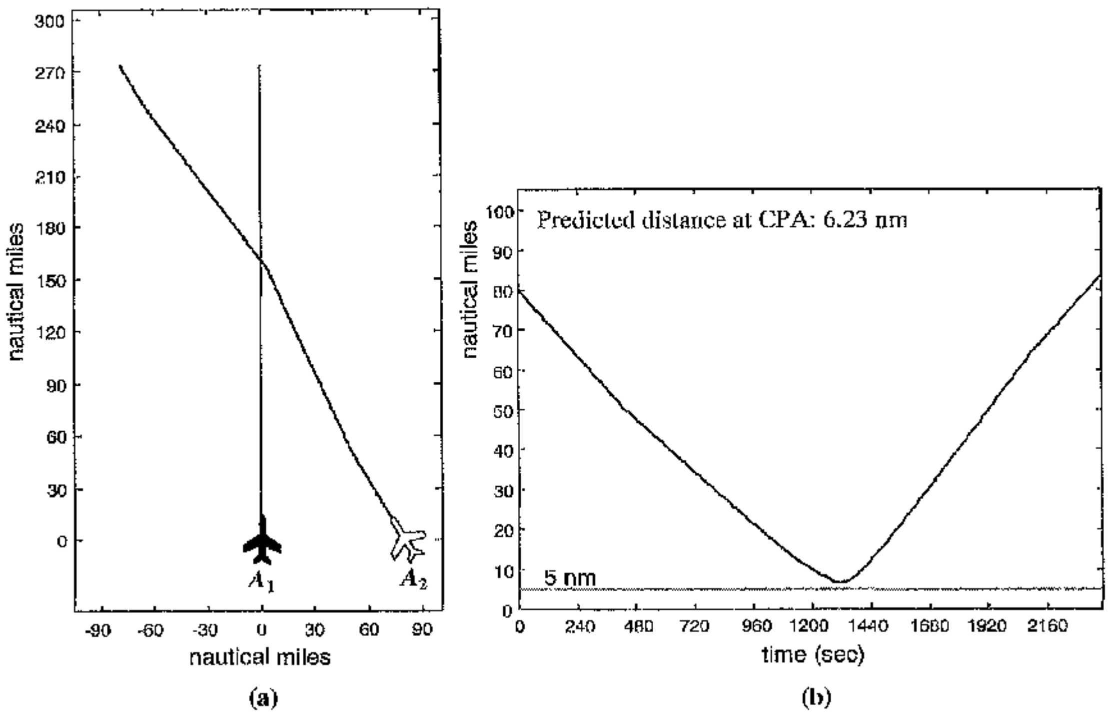


Figure 3.14. Simulation of the resolution of the conflict in scenario 1: A_2 acts as the leader and its flight crew accepts the proposed resolution trajectory 60 s after the detection of the conflict. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

	γ	t_s	t_e	Cost
A_1	Initially intended route			0.00
A_2	6^0	420 s	2090 s	2.77
Total conflict resolution cost				2.77

Table 3.5: Values of parameters and costs of the resolution trajectories in Figure 3.18(a)

3.4.5.5 Statistical analysis of the performance of the co-operation mechanism

Figure 3.11(a) above shows the resolution trajectories that result from simulating the application of the co-operation mechanism in scenario 1 assuming that A_1 acts as the leader and its flight crew accepts the proposed trajectory 60 seconds after conflict detection. Since these resolution trajectories are generated through a probabilistic iterative improvement process, they differ each time the simulation is run. The probabilistic nature of the trajectory-planning algorithm and the randomness of the flight crew response latency influence the performance of the co-operation mechanism.

Given these random elements, the resolution trajectories that result from applying the co-operation mechanism in a specific conflicting configuration are, albeit guaranteed to be conflict-free, neither unique for that configuration nor predictable *a priori*. Hence, to analyse the performance of the co-operation mechanism, the outcome of its application in a given conflict scenario will be studied from a statistical perspective. Four random variables have been defined to describe the performance of the co-operation mechanism in a particular conflict scenario. The four random variables are the cost of each of the resolution trajectories that result from applying the co-operation mechanism to the scenario, the sum of these two costs and the predicted minimum distance between the aircraft as they fly their resolution trajectories. These random variables are denoted as $\boldsymbol{c}_1$, $\boldsymbol{c}_2$, $\boldsymbol{s}_c$ and $\boldsymbol{d}_{A1\text{-}A2}$, respectively. Each of these random variables has an associated probability density function (PDF). The PDFs of the random variables $\boldsymbol{c}_1$, $\boldsymbol{c}_2$, $\boldsymbol{s}_c$ and $\boldsymbol{d}_{A1\text{-}A2}$ are denoted as $f(c_1)$, $f(c_2)$, $f(s_c)$, and $f(d_{A1\text{-}A2})$, respectively. In this notation, the PDF under consideration is identified by the independent variable of the function f.

The PDFs of the four random variables introduced above reflect not only the probabilistic nature of the algorithm but also the randomness of the response latencies of the flight crews. Thus, the four random variables $\boldsymbol{c}_1$, $\boldsymbol{c}_2$, $\boldsymbol{s}_c$ and $\boldsymbol{d}_{A1\text{-}A2}$ can be seen as related to the random variables $\boldsymbol{t}_1$, which models A_1's flight crew response latency and $\boldsymbol{t}_2$, which models A_2's flight crew response latency. Four three-dimensional random vectors can be defined, each of them including one of the four random variables introduced above together with the two random variables $\boldsymbol{t}_1$ and $\boldsymbol{t}_2$. Each of the random vectors is characterised by a joint probability density function [92]. For example, the random vector $(\boldsymbol{c}_1, \boldsymbol{t}_1, \boldsymbol{t}_2)$ has the associated PDF $f(c_1, t_1, t_2)$.

To simplify the analysis of the joint PDFs introduced above, the two possible sequences of actions for a given encounter will be considered separately. Assuming a specific sequence of actions, a conditional joint probability density function can be defined for each of the random vectors. Accordingly, given the two events:

$$E_1 = \{ A_1 \text{ acts as the leader } \} \text{ and } E_2 = \{ A_2 \text{ acts as the leader } \} \tag{3.23}$$

the following two conditional joint probability density functions can be defined for the random vector $(\boldsymbol{c}_1, \boldsymbol{t}_1, \boldsymbol{t}_2)$:

$$f(c_1, t_1, t_2 | \mathrm{E}_1) \text{ and } f(c_1, t_1, t_2 | \mathrm{E}_2) \tag{3.24}$$

Analogous conditional joint probability density functions can be defined for the random vectors (c_2, t_1, t_2), (s_c, t_1, t_2) and ($d_{A1\text{-}A2}$, t_1, t_2). The resolution trajectories that result from the application of the co-operation mechanism do not depend on the follower's flight crew response latency, providing that a sequence of actions has been specified. Hence, assuming a specific sequence of actions, each of the four random variables c_1, c_2, s_c and $d_{A1\text{-}A2}$ can be seen as related solely to the random variable modelling the response latency of the leader's flight crew. Consequently, the two conditional PDFs in (3.24) can be recast, respectively, as:

$$f(c_1, t_1 | \mathrm{E}_1) \text{ and } f(c_1, t_2 | \mathrm{E}_2) \tag{3.25}$$

In this context, the separate PDFs of each of the random variables c_1, c_2, s_c and $d_{A1\text{-}A2}$ assuming a specific sequence of actions can be seen as marginal density functions [92]. For example, the PDF of c_1 assuming that A_1 acts as the leader can be expressed as the marginal density of c_1 given the two-dimensional conditional probability density function $f(c_1, t_1 | \mathrm{E}_1)$. This marginal density is denoted as $f(c_1 | \mathrm{E}_1)$ and is related to $f(c_1, t_1 | \mathrm{E}_1)$ as follows:

$$f(c_1 | \mathrm{E}_1) = \int_{-\infty}^{+\infty} f(c_1, t_1 | \mathrm{E}_1)\, dt_1 \tag{3.26}$$

Similarly, the PDF of c_1 assuming that A_2 acts as the leader can be expressed as the marginal density of c_1 given the two-dimensional conditional probability density function $f(c_1, t_2 | \mathrm{E}_2)$. This marginal density is denoted as $f(c_1 | \mathrm{E}_2)$ and is related to $f(c_1, t_2 | \mathrm{E}_2)$ as follows:

$$f(c_1 | \mathrm{E}_2) = \int_{-\infty}^{+\infty} f(c_1, t_2 | \mathrm{E}_2)\, dt_2 \tag{3.27}$$

Since the two events E_1 and E_2 are mutually exclusive, an expression for the PDF of c_1 for a given conflict can be obtained applying the Total Probability Theorem [92]:

$$f(c_1) = f(c_1 | \mathrm{E}_1) \cdot \mathrm{P}(\mathrm{E}_1) + f(c_1 | \mathrm{E}_2) \cdot \mathrm{P}(\mathrm{E}_2) \tag{3.28}$$

where $P(E_1)$ denotes the probability of the event E_1 and $P(E_2)$ denotes the probability of the event E_2. The integral of the PDF in (3.28) over a given interval is the probability that the value of the random variable c_1 obtained when the co-operation mechanism is applied to the conflict scenario considered falls inside that interval regardless of which aircraft happens to act as the leader.

Assuming that the two aircraft detect the conflict simultaneously and considering that the random variables t_1 and t_2 are mutually independent, the probability of the event E_1 can be obtained as follows [92]:

$$\begin{aligned} P(E_1) &= P(t_1 < t_2) = P(z = t_2 - t_1 > 0) = \\ &= \int_0^{+\infty} f(z)\,dz = \int_0^{+\infty}\int_{-\infty}^{+\infty} f_{t_1 t_2}(t_1, z+t_1)\,dt_1\,dz = \\ &= \int_0^{+\infty}\int_{-\infty}^{+\infty} f_t(t_1)\, f_t(z+t_1)\,dt_1\,dz = 0.5 \end{aligned} \tag{3.29}$$

where $f_{t_1 t_2}$ denotes the joint probability density function $f(t_1, t_2)$ and f_t denotes the probability density function associated to both random variables t_1 and t_2, which is shown in Figure 3.6. For a generic real random variable t, f_t is given by:

$$f_t(t) = \begin{cases} 0 & \text{if } t \notin [40, 80] \\ \dfrac{1}{40} & \text{if } t \in [40, 80] \end{cases} \tag{3.30}$$

Analogously, the probability of A_2 acting as the leader can also be shown to be equal to 0.5. Hence, the two possible sequences of actions are equally likely. Considering the result in (3.28), the expression for the PDF of c_1 in (3.28) can be recast as follows:

$$f(c_1) = f(c_1 \mid E_1)\cdot 0.5 + f(c_1 \mid E_2)\cdot 0.5 \tag{3.31}$$

Expressions analogous to (3.31) can be obtained for the PDF of each of the random variables c_2, s_c and $d_{A1\text{-}A2}$.

The statistical analysis of the random variables c_1, c_2, s_c and $d_{A1\text{-}A2}$ in a given scenario is based on the repeated simulation of the application of the co-operation mechanism in that scenario assuming a specific sequence of actions. Once the roles of leader and follower have been assigned, the application of the mechanism is simulated n times for the scenario considered. In each simulation run, the response latency of the leader aircraft's flight crew is selected at random from the interval [40 s, 80 s]. The n simulation runs result in a sample of size n of each of the four random variables considered. The samples are drawn according to the conditional PDFs of the random variables assuming the specified sequence of actions. Providing that n is sufficiently large, the samples obtained can be used to make inferences about the conditional PDFs according to which they have been drawn. The Central Limit Theorem ([93], [92]) is applied to estimate the mean of the conditional PDF based on the sample mean. Considering that the PDFs of the four random variables c_1, c_2, s_c and $d_{A1\text{-}A2}$ can be expressed in the form of (3.31), the inferences made about their conditional PDFs can be used to make inferences about their PDFs.

Ott and Mendenhall state the Central Limit Theorem as follows:

> "If random samples containing a fixed number n of measurements are repeatedly drawn from a population with finite mean μ and standard deviation σ, then if n is large, the sample means will have a distribution that is approximately normal with mean $\mu_{\bar{x}} = \mu$ and standard deviation $\sigma_{\bar{x}} = \frac{\sigma}{\sqrt{n}}$." [93]

It is accepted that, in general, the Central Limit Theorem holds for n>30 [93]. If the size of a sample from a distribution is greater than 30, then the mean of that sample, $\bar{x}$, can be considered as a point estimate of the mean of the distribution from which the sample have been drawn. Since the sampling distribution for $\bar{x}$ is approximately normal with mean μ and standard deviation $\sigma_{\bar{x}}$, the interval $[\mu - 1.96\sigma_{\bar{x}}, \mu + 1.96\sigma_{\bar{x}}]$ would include 95% of the sample means in repeated sampling. If $\bar{x}$ lies in the interval $[\mu - 1.96\sigma_{\bar{x}}, \mu + 1.96\sigma_{\bar{x}}]$, which occurs with probability 0.95, the interval $[\bar{x} - 1.96\sigma_{\bar{x}}, \bar{x} + 1.96\sigma_{\bar{x}}]$ contains the parameter μ. Therefore, 95% of the time in repeated sampling, the interval $[\bar{x} - 1.96\sigma_{\bar{x}}, \bar{x} + 1.96\sigma_{\bar{x}}]$ contains the mean of the PDF,

μ. This interval is called a *95% confidence interval* and is shown in Figure 3.15. If σ is unknown, the sample standard deviation s can be substituted for σ in the formula for $\sigma_{\bar{x}}$ providing that n is reasonably large; the n>30 criterion is considered sufficient for mound-shaped PDFs [93].

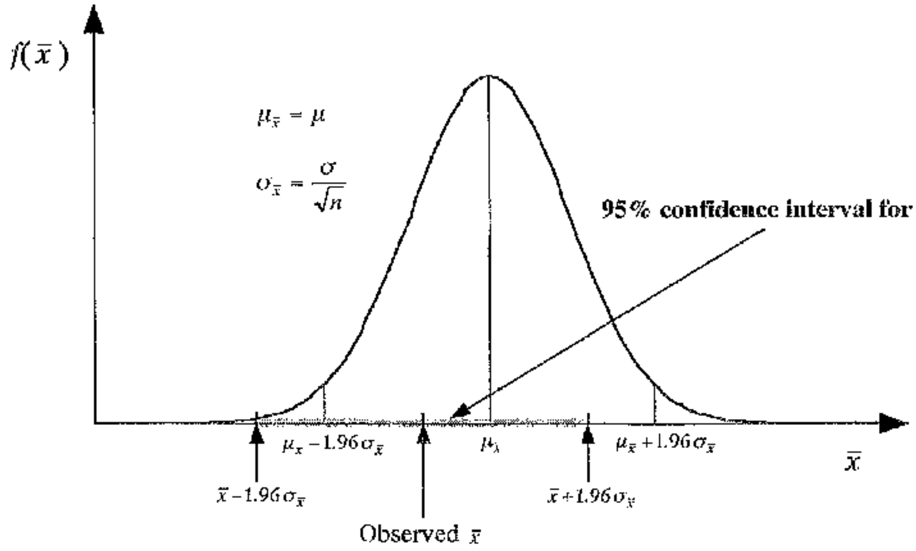
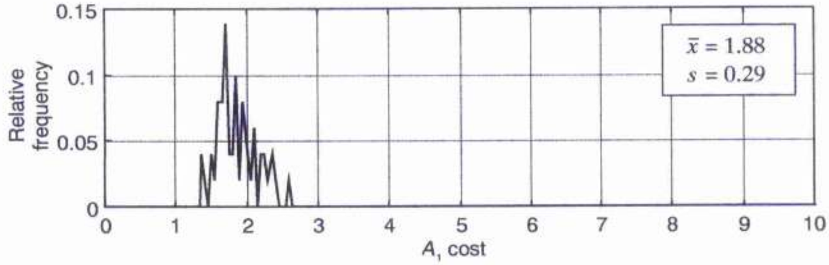


Figure 3.15. Interval estimate of the mean of a PDF based on the mean of a sample

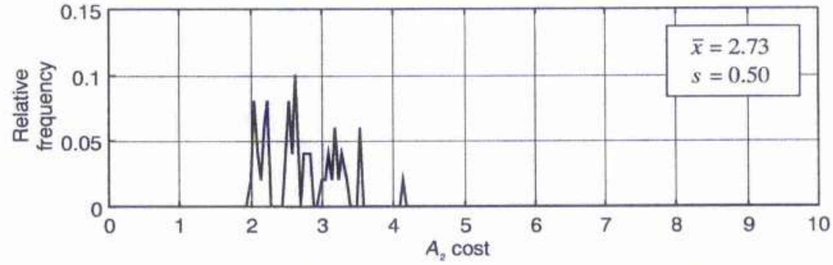
To allow for the use of the sample mean and standard deviation to make inferences about the PDFs, the value n = 50 has been chosen as the sample size for the simulations.

3.4.5.6 Statistical analysis of the performance of the co-operation mechanism in conflict scenario 1

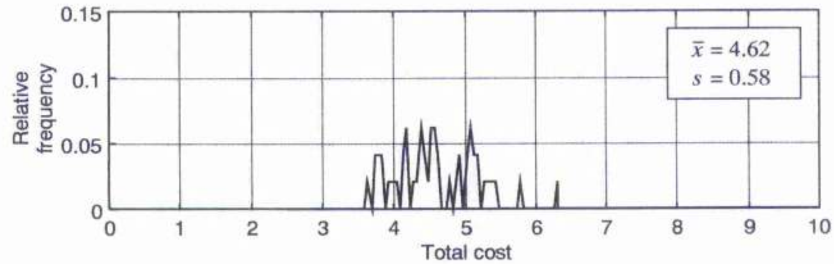
The application of the co-operation mechanism in scenario 1 has been simulated 50 times assuming that A_1 acts as the leader in all the simulation runs. In each simulation run, A_1's flight crew response latency has been selected at random from the interval [40 s, 60 s], according to the probability density function in Figure 3.6. The simulation runs result in a random sample of size 50 from each of the four random variables c_1, c_2, s_c and $d_{A1\text{-}A2}$. The samples obtained are drawn according to the conditional probability density functions $f(c_1|\ E_1)$, $f(c_2|\ E_1)$, $f(s_c|\ E_1)$ and $f(d_{A1\text{-}A2}|\ E_1)$, respectively. The relative frequency graphs of the samples obtained as well as their statistics are shown in Figure 3.16.



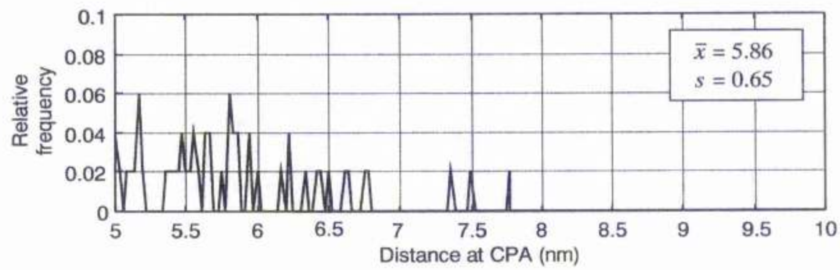
(a) Relative frequency graph and sample statistics for the cost of A_1's resolution trajectory



(b) Relative frequency graph and sample statistics for the cost of A_2's resolution trajectory



(c) Relative frequency graph and sample statistics for the total cost of the conflict resolution (cost of A_1's resolution trajectory + cost of A_2's resolution trajectory)



(d) Relative frequency graph and sample statistics for the predicted distance between the aircraft at the CPA

Figure 3.16. Simulation of the resolution of the conflict in scenario 1: 50 simulation runs; A_1 acts as the leader in all the simulation runs; in each simulation run, A_1's flight crew response latency is selected at random from the interval [40 s, 80s]

Each time a simulation is run, the trajectory-planning algorithm on board A_1 produces a resolution trajectory for the conflicting configuration in scenario 1. Since it is assumed that A_1 acts as the leader, this trajectory is independent of A_1's flight crew response time. Therefore, the PDF $f(c_1|\ \mathrm{E}_1)$, according to which the sample of the random variable $\boldsymbol{c_1}$ has been drawn, exclusively reflects the probabilistic nature of the algorithm. However, due to the variability of A_1's resolution trajectory and of the time at which A_2 receives that trajectory, in each simulation run the algorithm on board A_2 faces a different conflict configuration and searches for a response to a different resolution trajectory. Hence, the PDFs $f(c_2|\ \mathrm{E}_1)$, $f(s_c|\ \mathrm{E}_1)$ and $f(d_{\mathrm{A1\text{-}A2}}|\ \mathrm{E}_1)$ reflect not only the probabilistic nature of the algorithm, but also the randomness of the response latency of A_1's flight crew.

Given the size of the samples obtained, the sample mean can be considered as a point estimate of the mean of the PDF according to which the sample has been drawn. Additionally, interval estimates for the mean of that PDF can be formulated based on the values of the sample mean $\bar{x}$ and the sample standard deviation s. For example, according to the statistics of the sample in Figure 3.16(a), a point estimate of the mean of the PDF $f(c_1|\ \mathrm{E}_1)$ is given by the sample mean $\bar{x}$ =1.88 and a 95% confidence interval for that mean would be as follows:

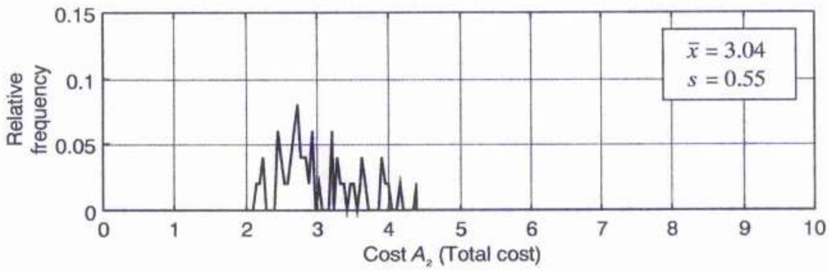
$$\left[\bar{x}-1.96\sigma_{\bar{x}}, \bar{x}+1.96\sigma_{\bar{x}}\right] \cong \left[\bar{x}-1.96s, \bar{x}+1.96s\right] = \left[1.31, 2.45\right] \tag{3.32}$$

If the confidence coefficient (0.95 for a 95% confidence interval) is reduced the amplitude of the confidence interval is also reduced. A 90% confidence interval (confidence coefficient 0.90) for the mean of the PDF of the cost of A_1's resolution trajectory based on the same sample would be as follows:

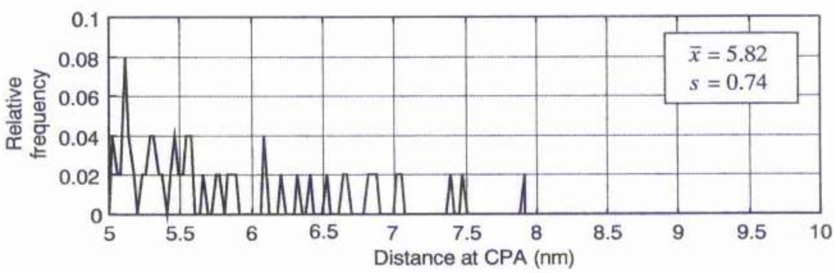
$$\left[\bar{x}-1.64\sigma_{\bar{x}}, \bar{x}+1.64\sigma_{\bar{x}}\right] \cong \left[\bar{x}-1.64s, \bar{x}+1.64s\right] = \left[1.40, 2.35\right] \tag{3.33}$$

Point estimates and confidence intervals for the means of the PDFs of $\boldsymbol{c_2}$, $\boldsymbol{s_c}$, and $\boldsymbol{d_{\mathrm{A1\text{-}A2}}}$ assuming A_1 acting as the leader can be formulated analogously.

To draw a random sample from each of the random variables $\boldsymbol{c_1}$, $\boldsymbol{c_2}$, $\boldsymbol{s_c}$ and $\boldsymbol{d_{\mathrm{A1\text{-}A2}}}$ according to the conditional PDFs $f(c_1|\ \mathrm{E}_2)$, $f(c_2|\ \mathrm{E}_2)$, $f(s_c|\ \mathrm{E}_2)$ and $f(d_{\mathrm{A1\text{-}A2}}|\ \mathrm{E}_2)$, respectively, the application of the co-operation mechanism in scenario 1 has been simulated 50 times assuming that A_2 acts as the leader. In each of the simulation runs, A_2's flight crew response latency has been selected at random from the interval [40 s, 60 s], according to the probability density function in Figure 3.6. The relative frequency graphs of the samples obtained as well as their statistics are shown in Figure 3.17.



(a) Relative frequency graph and sample statistics for the cost of A_2's resolution trajectory (total cost)



(d) Relative frequency graph and sample statistics for the predicted distance between the aircraft at the CPA

Figure 3.17. Simulation of the resolution of the conflict in scenario 1: 50 simulation runs; A_2 acts as the leader in all the simulation runs; in each simulation run, A_2's flight crew response latency is selected at random from the interval [40 s, 80s]

Analogously to the case in which A_1 is assumed to be the leader, the conditional PDF $f(c_2|\ E_2)$ exclusively reflects the probabilistic nature of the algorithm. In all the simulation runs A_1 maintains its initially intended route, which does not conflict with A_2's resolution trajectory.

According to the statistics of the sample in Figure 3.17(a), a point estimate of the mean of the PDF $f(c_2|\ E_2)$ is given by the sample mean $\bar{x}$ =3.04 and a 95% confidence interval for that mean is given by [1.96, 4.11]. Similarly, the sample mean $\bar{x}$ =5.82 nm is a point estimate of the mean of the PDF $f(d_{A1\text{-}A2}|\ E_2)$ and the interval [4.34 nm, 7.25 nm] is a 95% confidence interval for that mean.

Considering that the PDFs of the random variables c_1, c_2, s_c and $d_{A1\text{-}A2}$ for a given conflict scenario can be expressed in the form of (3.31), point estimates of the means of those PDFs can be formulated based on the means of the samples obtained from the simulations. Thus, a point estimate for the mean of the PDF $f(c_1)$ can be obtained as follows:

Point estimate of the mean of $f(c_1)$:

$$0.5 \cdot \bar{x}(\text{sample from } f(c_1|E_1)) + 0.5 \cdot \bar{x}(\text{sample from } f(c_1|E_2)) = 0.94 \qquad (3.34)$$

where $\bar{x}(\text{sample from } f(c_1|E_1))$ and $\bar{x}(\text{sample from } f(c_1|E_2))$ denote the means of the samples obtained from the conditional PDFs $f(c_1|\ E_1)$ and $f(c_1|\ E_2)$, respectively. Point estimates can be formulated analogously for the PDFs of the other random variables considered:

Point estimate of the mean of $f(c_2)$:

$$0.5 \cdot \bar{x}(\text{sample from } f(c_2|E_1)) + 0.5 \cdot \bar{x}(\text{sample from } f(c_2|E_2)) = 2.88 \qquad (3.35)$$

Point estimate of the mean of $f(s_c)$:

$$0.5 \cdot \bar{x}(\text{sample from } f(s_c|E_1)) + 0.5 \cdot \bar{x}(\text{sample from } f(s_c|E_2)) = 3.83 \qquad (3.36)$$

Point estimate of the mean of $f(d_{A1\text{-}A2})$:

$$0.5 \cdot \bar{x}(\text{sample from } f(d_{A1-A2}|E_1)) + 0.5 \cdot \bar{x}(\text{sample from } f(d_{A1-A2}|E_2)) = 5.84 \qquad (3.37)$$

According to the point estimates above, the application of the co-operation mechanism results in the distribution of the total resolution cost between the two conflicting aircraft, with A_2's share of that cost expected to be higher than A_1's share.

3.4.5.7 Application of the co-operation mechanism in scenario 2

The application of the co-operation mechanism in scenario 2, which is depicted in Figure 3.18, has been simulated to illustrate the performance of the mechanism in a different conflicting configuration. It is assumed that the ADS-B range of coverage in scenario 2 is the same as in scenario 1. Consequently, as it can be observed in Figures 3.10(a) and 3.18(a), the distance between the two aircraft when the conflict is detected, d_d, is the same in both scenarios. A_1's speed, heading and initial position are the same in both scenarios. Aircraft A_2 is flying at the same speed in both scenarios but its heading and initial position in scenario 2 are different from those in scenario 1. Comparing Figures 3.10(b) and 3.18(b), it can be seen that the time lag between the detection of the conflict and the predicted closest point of approach is much shorter in scenario 2 than in scenario 1. This difference between the two scenarios will be shown to cause differences in the performance of the co-operation mechanism.

The application of the co-operation mechanism in scenario 2 has been simulated assuming that A_1 acts as the leader and its flight crew accepts the proposed resolution trajectory 60 seconds after conflict detection. Figure 3.19 shows the result of this simulation and Table 3.6 displays the parameters defining the two resolution trajectories in Figure 3.19(a) as well as their associated costs.

Figure 3.20 shows the result of simulating the application of the co-operation mechanism in scenario 2 assuming that A_2 acts as the leader and its flight crew accepts the proposed resolution trajectory 60 seconds after it detects the conflict. Table 3.7 shows the parameters defining the resolution trajectories in Figure 3.20(a) as well as their associated costs.

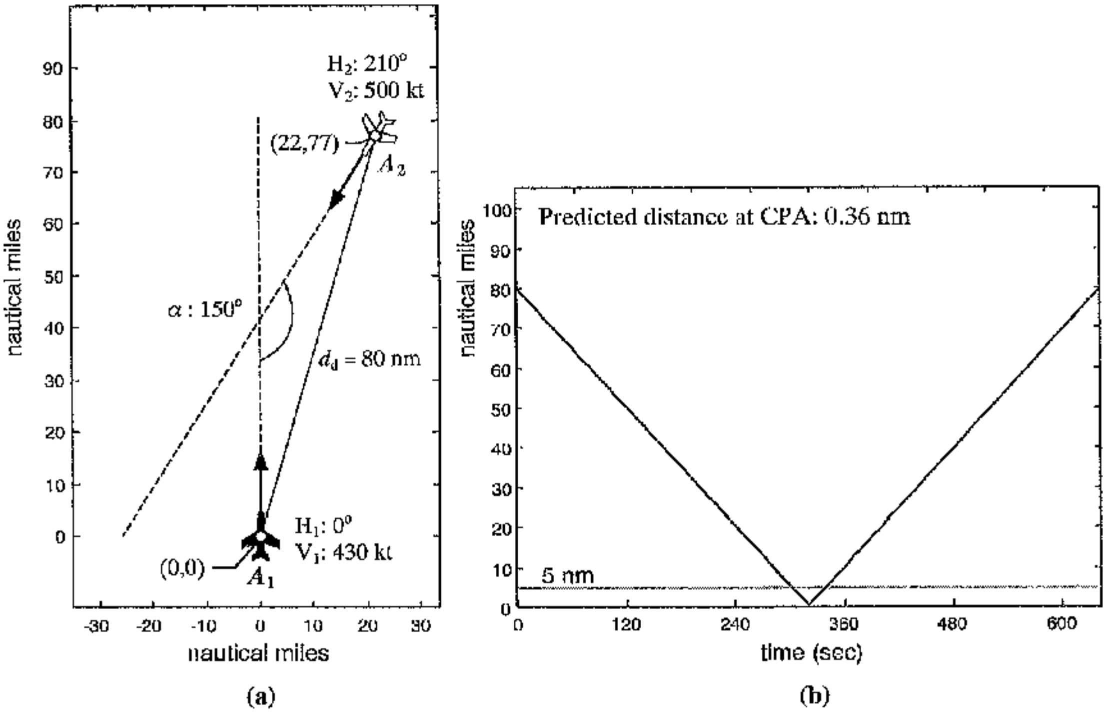


Figure 3.18. Conflict scenario 2. (a) Initial configuration (b) Predicted distances between the aircraft as they fly along their initially intended routes

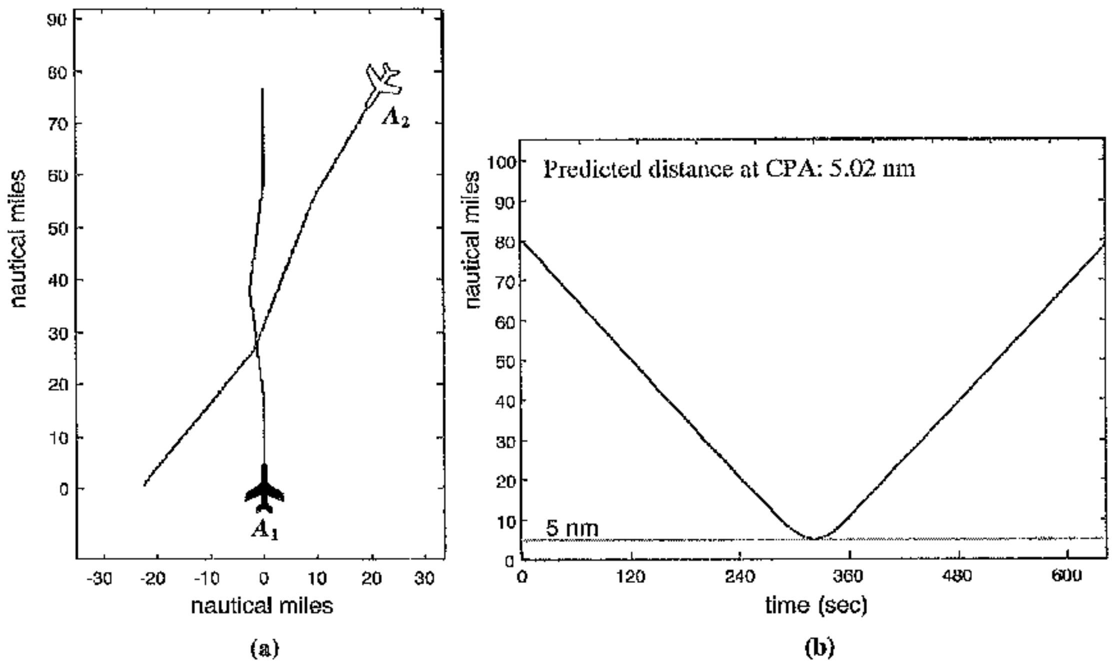


Figure 3.19. Simulation of the resolution of the conflict in scenario 2: A_1 acts as the leader and its flight crew accepts the proposed resolution trajectory 60 s after the detection of the conflict. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

	γ	t_s	t_e	Cost
A_1	-7°	151 s	486 s	2.05
A_2	-9°	181 s	630 s	3.02
Total conflict resolution cost				5.07

Table 3.6: Values of parameters and costs of the resolution trajectories in Figure 3.19(a)

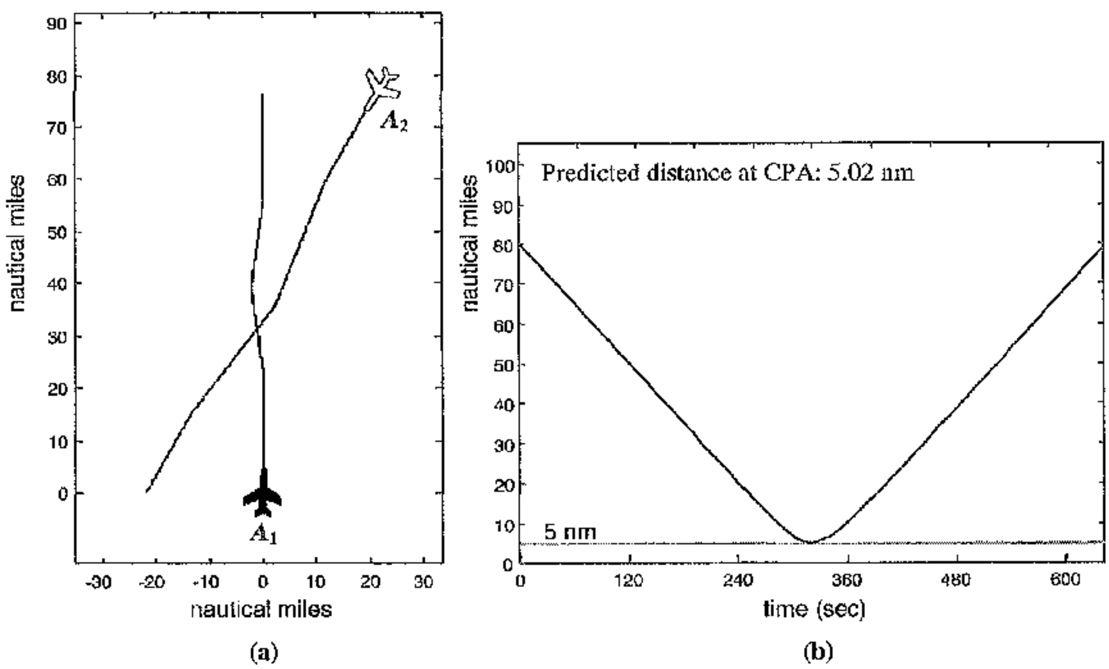


Figure 3.20. Simulation of the resolution of the conflict in scenario 2: A_2 acts as the leader and its flight crew accepts the proposed resolution trajectory 60 s after the detection of the conflict. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

	γ	t_s	t_e	Cost
A_1	-8°	195 s	466 s	2.31
A_2	-8°	145 s	515 s	2.50
Total conflict resolution cost				4.81

Table 3.7: Values of parameters and costs of the resolution trajectories in Figure 3.20(a)

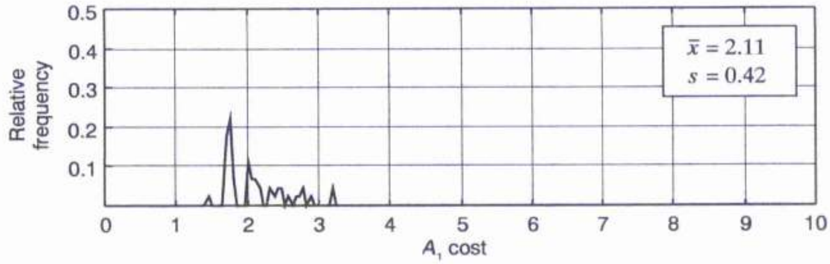
3.4.5.8 Statistical analysis of the performance of the co-operation mechanism in scenario 2

The methodology used in section 3.4.5.6 to statistically analyse the performance of the co-operation mechanism in scenario 1 is used in this section for scenario 2. The application of the co-operation mechanism in scenario 2 has been simulated 50 times assuming that A_1 acts as the leader in all the simulation runs. In each of the simulation runs, A_1's flight crew response latency has been selected at random from the interval [40 s, 60 s], according to the probability density function in Figure 3.6.

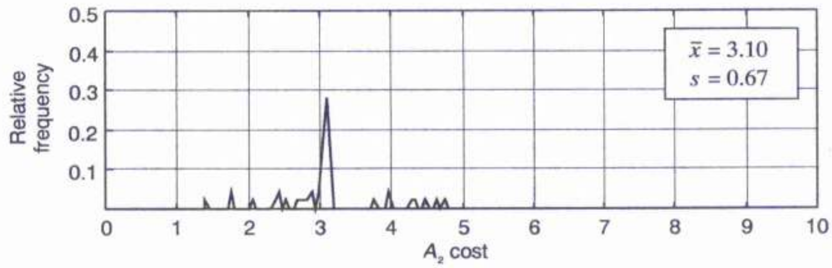
The simulation runs result in a random sample of size 50 from each of the four random variables c_1, c_2, s_c and $d_{A1\text{-}A2}$. The samples obtained are drawn according to the conditional probability density functions $f(c_1|\ E_1)$, $f(c_2|\ E_1)$, $f(s_c|\ E_1)$ and $f(d_{A1\text{-}A2}|\ E_1)$, respectively. The relative frequency graphs of the samples obtained as well as their statistics are shown in Figure 3.21. The PDF $f(c_1|\ E_1)$ exclusively reflects the probabilistic nature of the algorithm on board A_1 in the conflicting configuration considered, while the PDFs $f(c_2|\ E_1)$, $f(s_c|\ E_1)$ and $f(d_{A1\text{-}A2}|\ E_1)$ reflect not only the probabilistic nature of the algorithm, but also the randomness of the response latency of A_1's flight crew.

Point estimates and confidence intervals for the means of the PDFs $f(c_1|\ E_1)$, $f(c_2|\ E_1)$, $f(s_c|\ E_1)$ and $f(d_{A1\text{-}A2}|\ E_1)$ can be formulated based on the statistics of the samples obtained. For example, a point estimate of the mean of the PDF $f(c_1|\ E_1)$ is given by the sample mean $\bar{x}=2.11$ and a 95% confidence interval for that mean is given by [1.29, 2.93].

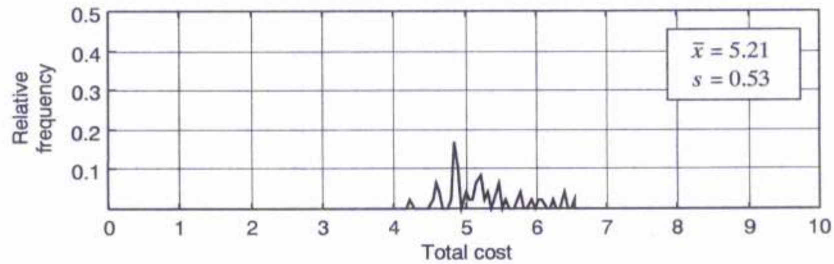
To draw a random sample from each of the random variables c_1, c_2, s_c and $d_{A1\text{-}A2}$ according to the conditional PDFs $f(c_1|\ E_2)$, $f(c_2|\ E_2)$, $f(s_c|\ E_2)$ and $f(d_{A1\text{-}A2}|\ E_2)$, respectively, the application of the co-operation mechanism in scenario 2 has been simulated 50 times assuming that A_2 acts as the leader. In each of the simulation runs, A_2's flight crew response latency has been selected at random from the interval [40 s, 60 s], according to the probability density function in Figure 3.6. The relative frequency graphs of the samples obtained as well as their statistics are shown in Figure 3.22.



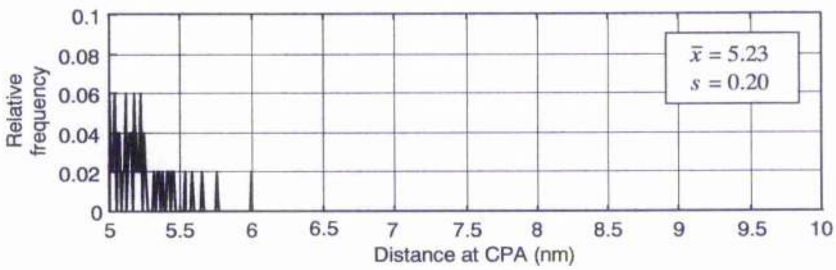
(a) Relative frequency graph and sample statistics for the cost of A_1's resolution trajectory



(b) Relative frequency graph and sample statistics for the cost of A_2's resolution trajectory

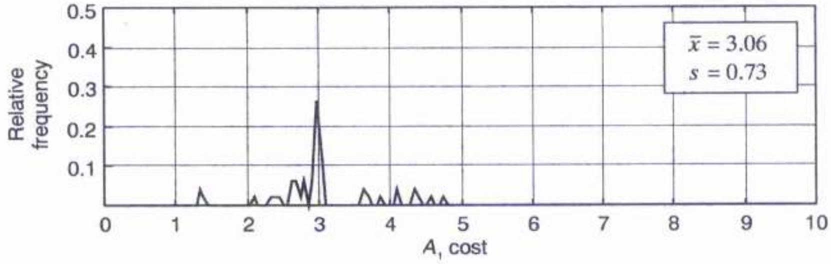


(c) Relative frequency graph and sample statistics for the total cost of the conflict resolution (cost of A_1's resolution trajectory + cost of A_2's resolution trajectory)

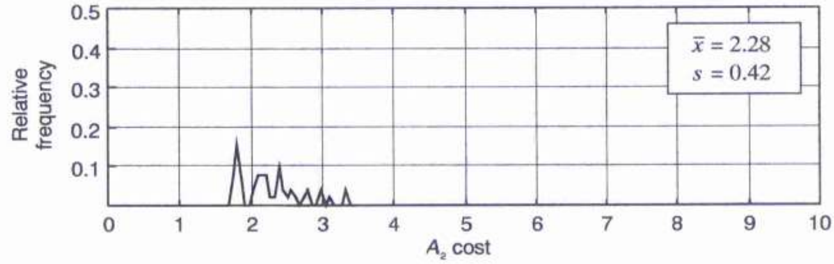


(d) Relative frequency graph and sample statistics for the predicted distance between the aircraft at the CPA

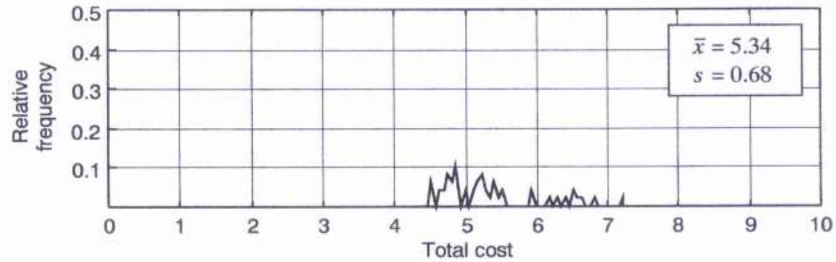
Figure 3.21. Simulation of the resolution of the conflict in scenario 2: 50 simulation runs; A_1 acts as the leader in all the simulation runs; in each simulation run, A_1's flight crew response latency is selected at random from the interval [40 s, 80s]



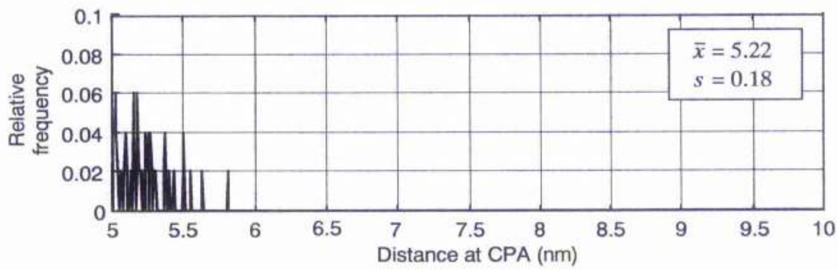
(a) Relative frequency graph and sample statistics for the cost of A_1's resolution trajectory



(b) Relative frequency graph and sample statistics for the cost of A_2's resolution trajectory



(c) Relative frequency graph and sample statistics for the total cost of the conflict resolution (cost of A_1's resolution trajectory + cost of A_2's resolution trajectory)



(d) Relative frequency graph and sample statistics for the predicted distance between the aircraft at the CPA

Figure 3.22. Simulation of the resolution of the conflict in scenario 2: 50 simulation runs; A_2 acts as the leader in all the simulation runs; in each simulation run, A_2's flight crew response latency is selected at random from the interval [40 s, 80s]

Considering that the PDFs of the random variables c_1, c_2, s_c and $d_{A1\text{-}A2}$ for scenario 2 can be expressed in the form of (3.31), point estimates of the means of those PDFs can be obtained as follows:

Point estimate of the mean of $f(c_1)$:

$$0.5 \cdot \bar{x}(\text{sample from } f(c_1|\mathrm{E}_1)) + 0.5 \cdot \bar{x}(\text{sample from } f(c_1|\mathrm{E}_2)) = 2.58 \tag{3.38}$$

Point estimate of the mean of $f(c_2)$:

$$0.5 \cdot \bar{x}(\text{sample from } f(c_2|\mathrm{E}_1)) + 0.5 \cdot \bar{x}(\text{sample from } f(c_2|\mathrm{E}_2)) = 2.69 \tag{3.39}$$

Point estimate of the mean of $f(s_c)$:

$$0.5 \cdot \bar{x}(\text{sample from } f(s_c|\mathrm{E}_1)) + 0.5 \cdot \bar{x}(\text{sample from } f(s_c|\mathrm{E}_2)) = 5.27 \tag{3.40}$$

Point estimate of the mean of $f(d_{A1\text{-}A2})$:

$$0.5 \cdot \bar{x}(\text{sample from } f(d_{A1-A2}|\mathrm{E}_1)) + 0.5 \cdot \bar{x}(\text{sample from } f(d_{A1-A2}|\mathrm{E}_2)) = 5.22 \tag{3.41}$$

According to the point estimates above, in this scenario the application of the co-operation mechanism results in the equitable distribution of the total resolution cost between the two conflicting aircraft.

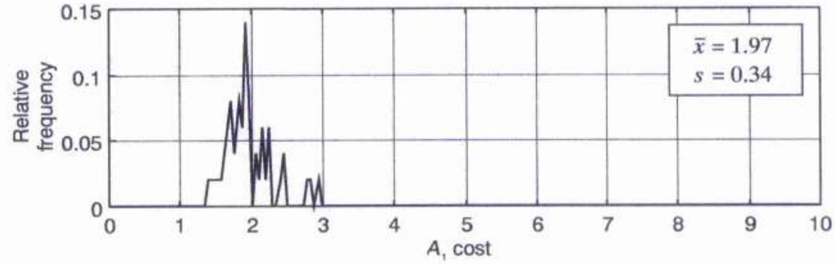
3.4.5.9 Comparison between the performance of the co-operation mechanism in scenarios 1 and 2

In this section, the differences in the performance of the co-operation mechanism caused by the shorter time lag between the detection of the conflict and the CPA in scenario 2 are highlighted. To do so, the performance of the co-operation mechanism in the two scenarios considered is statistically analysed assuming a specific sequence of actions and different fixed values of the response latency of the leader's flight crew.

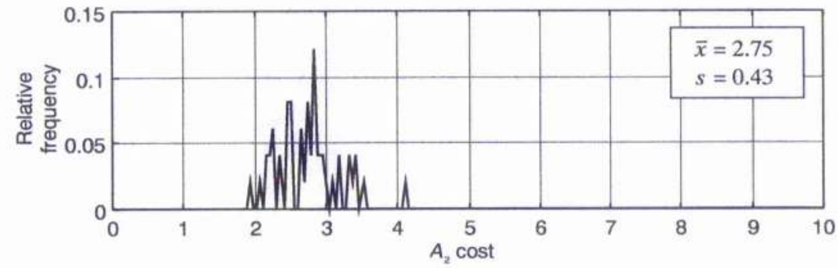
The application of the mechanism in scenario 1 has been repeatedly simulated 50 times assuming that A_1 acts as the leader and that A_1's flight crew response latency is 60 seconds in all simulation runs. This response time coincides with the mean of the PDF of the flight crew response latency (see Figure 3.6). The simulations result in samples of size 50 from each of the random variables $\boldsymbol{c}_1$, $\boldsymbol{c}_2$, $\boldsymbol{s}_c$ and $\boldsymbol{d}_{A1\text{-}A2}$. Figure 3.23 shows the relative frequency graphs of the samples obtained as well as their statistics. In this case, the samples have been drawn according to the conditional probability density functions $f(c_1|\ \mathrm{E}_1,\ t_1 = 60)$, $f(c_2|\ \mathrm{E}_1,\ t_1 = 60)$, $f(s_c|\ \mathrm{E}_1,\ t_1 = 60)$ and $f(d_{A1\text{-}A2}|\ \mathrm{E}_1,\ t_1 = 60)$, respectively. To illustrate the performance of the mechanism with a long response latency of the leader's flight crew, the application of the mechanism has been simulated 50 times assuming that A_1 acts as the leader and its flight crew's response latency is 75 seconds. The relative frequency graphs of the samples obtained as well as their statistics are shown in Figure 3.24. In this case, the samples have been drawn according to the conditional probability density functions $f(c_1|\ \mathrm{E}_1,\ t_1 = 75)$, $f(c_2|\ \mathrm{E}_1,\ t_1 = 75)$, $f(s_c|\ \mathrm{E}_1,\ t_1 = 75)$ and $f(d_{A1\text{-}A2}|\ \mathrm{E}_1,\ t_1 = 75)$. Comparing Figure 3.23 with Figure 3.24, it can be said that the performance of the algorithm in scenario 1 is not affected appreciably when the leader aircraft's response latency increases.

The application of the mechanism in scenario 2 has also been repeatedly simulated 50 times assuming that A_1 acts as the leader and its flight crew response latency is 60 seconds and other 50 times assuming that A_1 acts as the leader and its flight crew response latency is 75 seconds. The relative frequency graphs and the statistics of the samples obtained in each of these two series of simulations are shown in Figures 3.25 and 3.26, respectively.

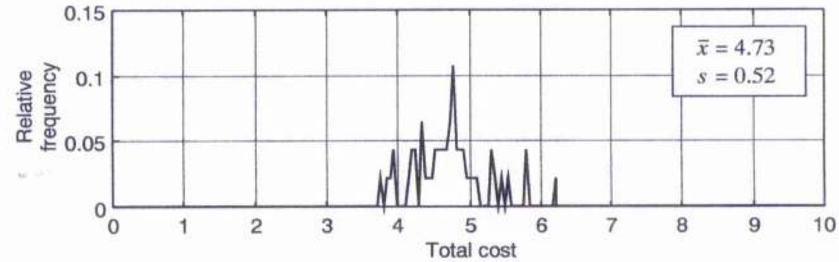
Comparing Figure 3.25(b) with Figure 3.26(b), it can be observed that the sample of the random variable $\boldsymbol{c}_2$ obtained for the 75 seconds response latency is distributed over a wider interval and contains greater values than the sample of that variable obtained for the 60 seconds response latency. Consequently, both the sample mean and the sample standard deviation are greater for the longer response latency. This shows that the performance of the co-operation mechanism in scenario 2 is heavily influenced by the leader's response latency. This performance variation, which was not observed in scenario 1, is due to the shorter time between the conflict detection and the CPA in



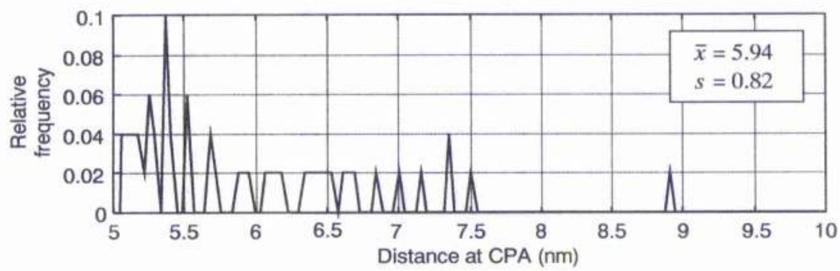
(a) Relative frequency graph and sample statistics for the cost of A_1's resolution trajectory



(b) Relative frequency graph and sample statistics for the cost of A_2's resolution trajectory

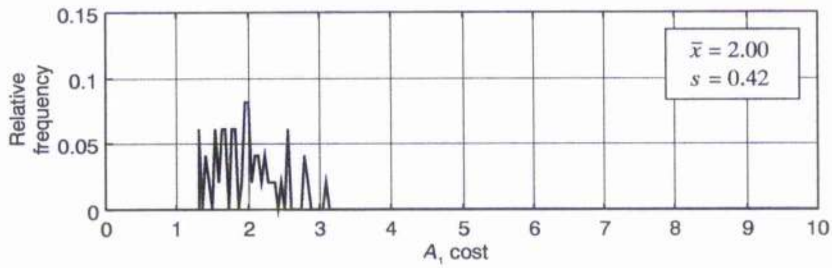


(c) Relative frequency graph and sample statistics for the total cost of the conflict resolution (cost of A_1's resolution trajectory + cost of A_2's resolution trajectory)

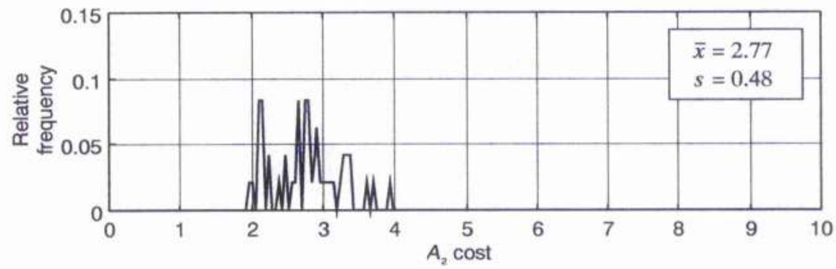


(d) Relative frequency graph and sample statistics for the predicted distance between the aircraft at the CPA

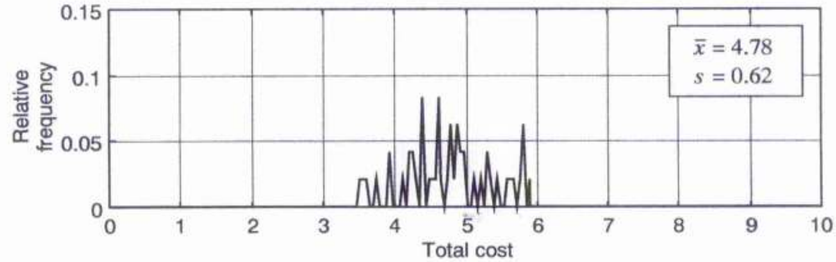
Figure 3.23. Probabilistic nature of the trajectory-planning algorithm. The simulation of the resolution process has been run 50 times for the conflict in scenario 1. In all the simulation runs A_1 acts as the leader and its flight crew accepts the proposed resolution trajectory 60 s after the detection of the conflict



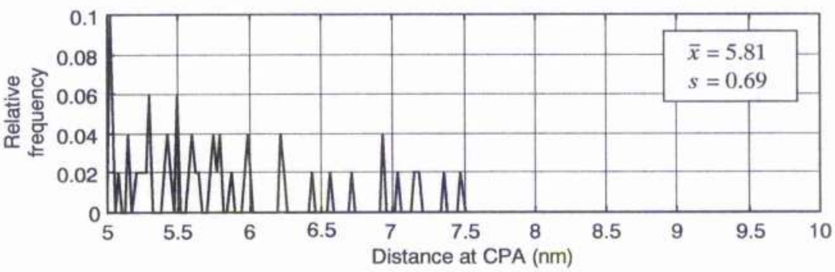
(a) Relative frequency graph and sample statistics for the cost of A_1's resolution trajectory



(b) Relative frequency graph and sample statistics for the cost of A_2's resolution trajectory

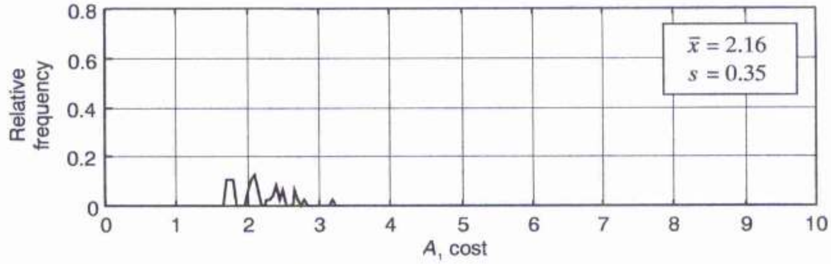


(c) Relative frequency graph and sample statistics for the total cost of the conflict resolution (cost of A_1's resolution trajectory + cost of A_2's resolution trajectory)

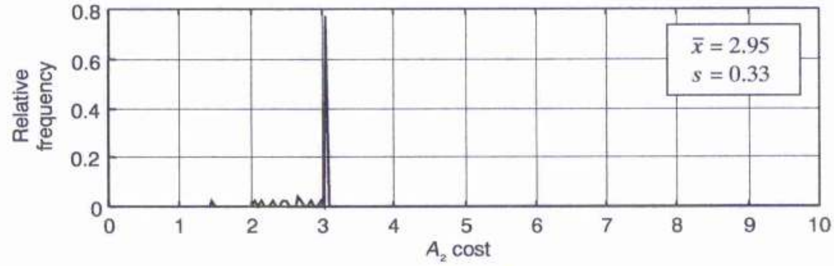


(d) Relative frequency graph and sample statistics for the predicted distance between the aircraft at the CPA

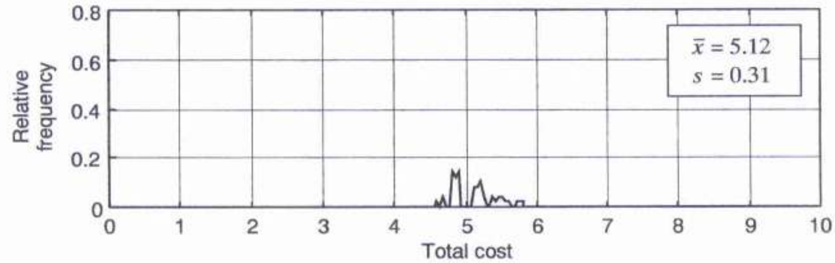
Figure 3.24. Probabilistic nature of the trajectory-planning algorithm and influence of the flight crew response latency. The simulation of the resolution process has been run 50 times for the conflict in scenario 1. In all the simulation runs A_1 acts as the leader and its flight crew accepts the proposed resolution trajectory 75 s after the detection of the conflict



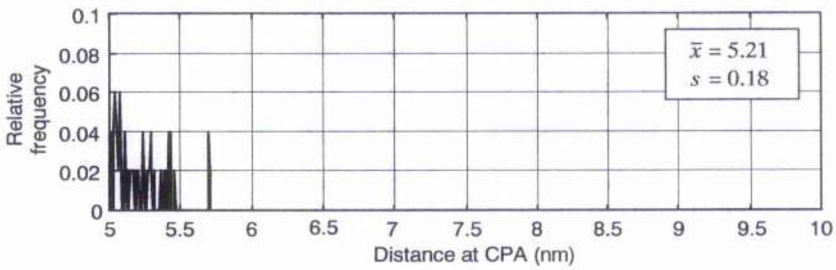
(a) Relative frequency graph and sample statistics for the cost of A_1's resolution trajectory



(b) Relative frequency graph and sample statistics for the cost of A_2's resolution trajectory

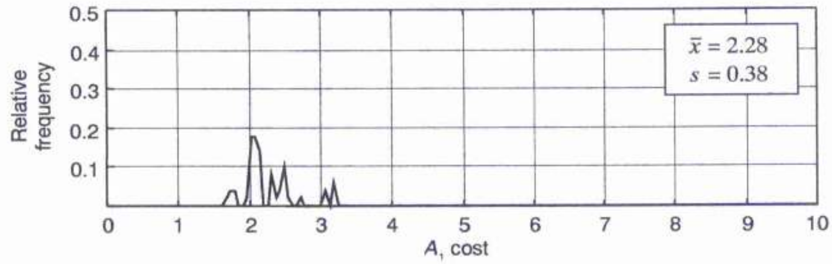


(c) Relative frequency graph and sample statistics for the total cost of the conflict resolution (cost of A_1's resolution trajectory + cost of A_2's resolution trajectory)

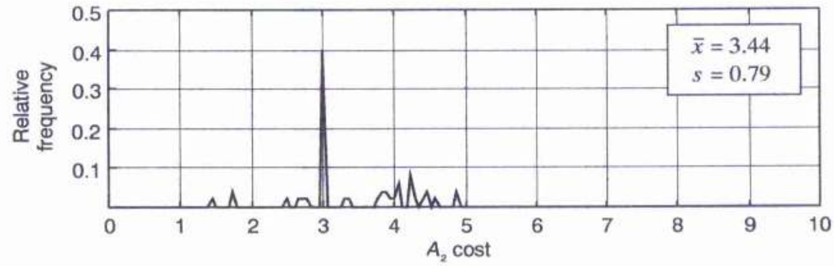


(d) Relative frequency graph and sample statistics for the predicted distance between the aircraft at the CPA

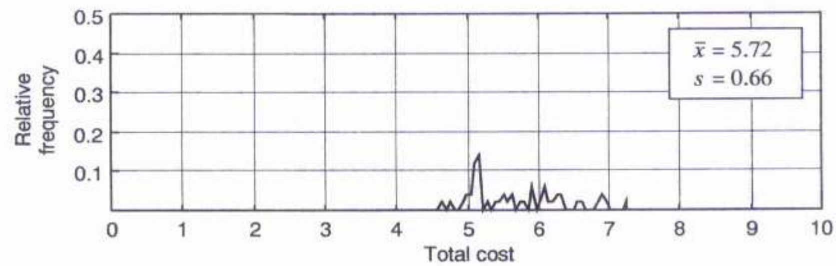
Figure 3.25. Probabilistic nature of the trajectory-planning algorithm. The simulation of the resolution process has been run 50 times for the conflict in scenario 2. In all the simulation runs A_1 acts as the leader and its flight crew accepts the proposed resolution trajectory 60 s after the detection of the conflict



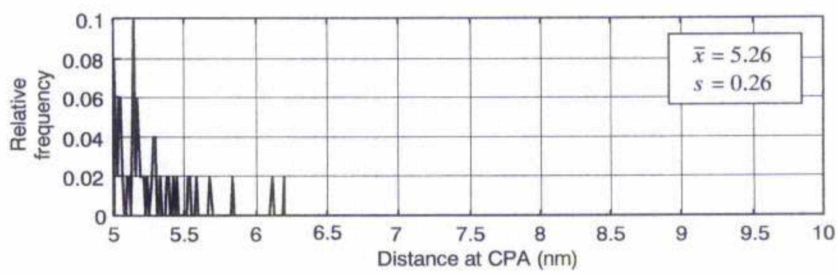
(a) Relative frequency graph and sample statistics for the cost of A_1's resolution trajectory



(b) Relative frequency graph and sample statistics for the cost of A_2's resolution trajectory



(c) Relative frequency graph and sample statistics for the total cost of the conflict resolution (cost of A_1's resolution trajectory + cost of A_2's resolution trajectory)



(d) Relative frequency graph and sample statistics for the predicted distance between the aircraft at the CPA

Figure 3.26. Probabilistic nature of the trajectory-planning algorithm and influence of the flight crew response latency. The simulation of the resolution process has been run 50 times for the conflict in scenario 2. In all the simulation runs A_1 acts as the leader and its flight crew accepts the proposed resolution trajectory 75 s after the detection of the conflict

t

scenario 2. In this scenario, the reduction of the time between the reception of the leader's resolution trajectory by the follower and their CPA caused by long response latencies of the leader may result in an increase in the cost of the follower's response. As the follower gets closer to the predicted CPA and the time available to resolve the conflict decreases, the response resolution trajectories produced by the algorithm may involve considerably more manoeuvring than in situations when the resolution trajectories are planned long before the CPA. The initial configuration in scenario 1 allows for longer times between the reception of the leader's resolution trajectory and the CPA than the initial configuration in scenario 2. Consequently, in scenario 1 the cost of A_2's resolution trajectory is not appreciably affected by variations in A_1's flight crew response latency.

An example of the application of the co-operation mechanism in scenario 2 assuming that A_1 acts as the leader and that its flight crew accepts the proposed resolution trajectory 75 seconds after the detection of the conflict is shown in Figure 3.27. Table 3.8 shows the parameters defining the resolution trajectories in Figure 3.27(a) as well as their associated costs.

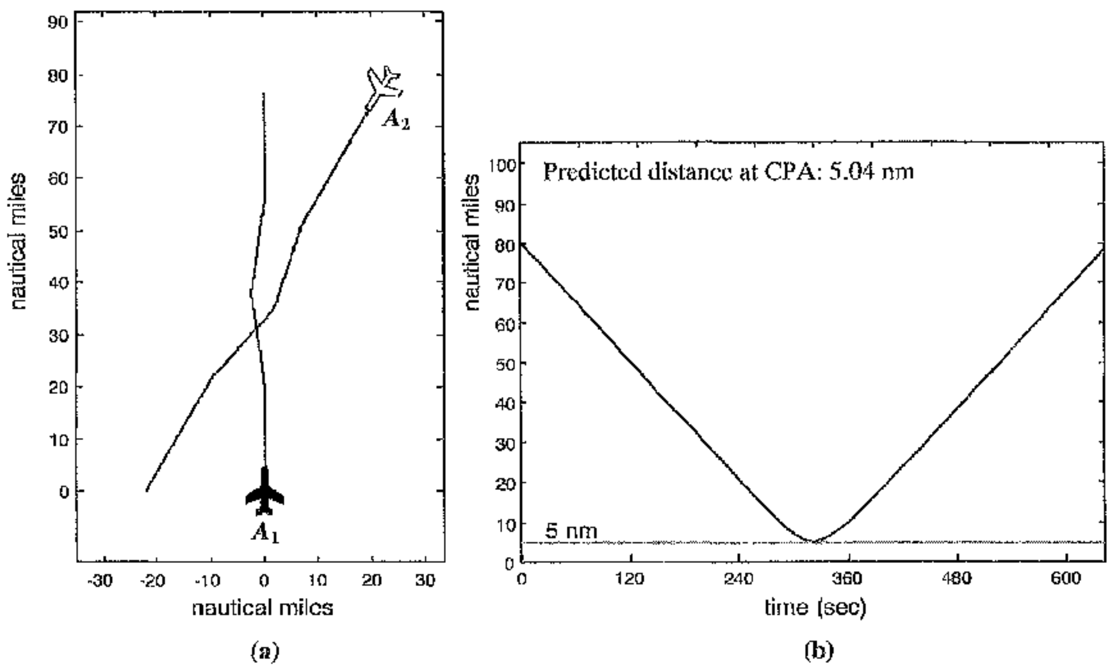


Figure 3.27. Simulation of the resolution of the conflict in scenario 2: A_1 acts as the leader and its flight crew accepts the proposed resolution trajectory 75 s after the detection of the conflict. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

	γ	t_s	t_e	Cost
A_1	-8°	173 s	467 s	2.34
A_2	-12°	216 s	461 s	3.74
Total conflict resolution cost				6.08

Table 3.8: Values of parameters and costs of the resolution trajectories in Figure 3.27(a)

Comparing the example in Figure 3.27 with the one presented earlier for a leader's response latency of 60 seconds (see Figure 19 and Table 3.6), it can be observed that A_2's resolution trajectory involves more significant heading changes with a response latency of 75 seconds.

3.4.5.10. Comparison of the performance of the co-operation mechanism in various conflicting configurations

In this section, the performance of the co-operation mechanism is analysed comparatively across different conflict scenarios. The analysis focuses on the variations in the costs of the resolution trajectories that result from applying the mechanism in scenarios with different values of the parameters V_1, V_2, d_d, d_{CPA} and α. In particular, these costs are compared across the different conflict scenarios that result from changing the value of α for given fixed values of V_1, V_2, d_d and d_{CPA}.

Once the values of the five parameters V_1, V_2, d_d, and d_{CPA} and α have been established, a conflict scenario can be fully defined by arbitrarily selecting the initial position and heading of aircraft A_1. The conflicting configuration in the resulting scenario does not depend on the initial position and heading chosen for A_1. All the conflict scenarios that result from choosing different initial positions and headings for A_1 correspond to the same conflicting configuration referred to different axes. To reduce the number of parameters, only conflicts with d_{CPA} = 0 nm will be considered. Given fixed values for V_1, V_2 and d_d, the performance of the co-operation mechanism is analysed comparatively between conflict scenarios with different values of α. With a view to establishing a common reference for the analysis, it is assumed that A_1's initial position coincides with the origin of the co-ordinate system and its heading is H_1 = 0° in all the conflict scenarios considered. Hence, given the values of the four parameters V_1, V_2 and d_d and α, the resulting conflict scenario is shown in Figure 3.28.

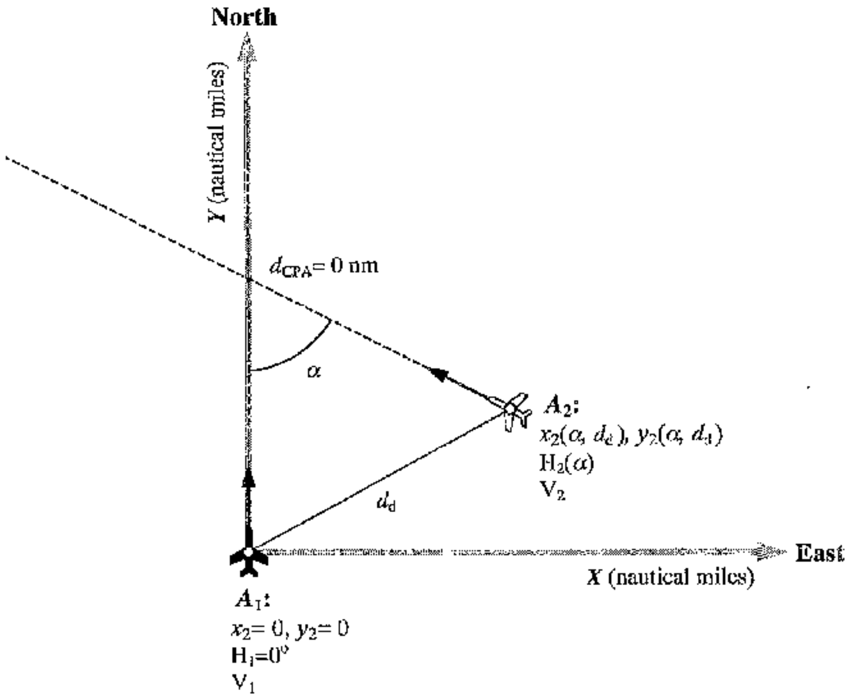


Figure 3.28. Conflict scenario for the analysis of the performance of the algorithm for different values of the parameters V_1, V_2, α and d_d

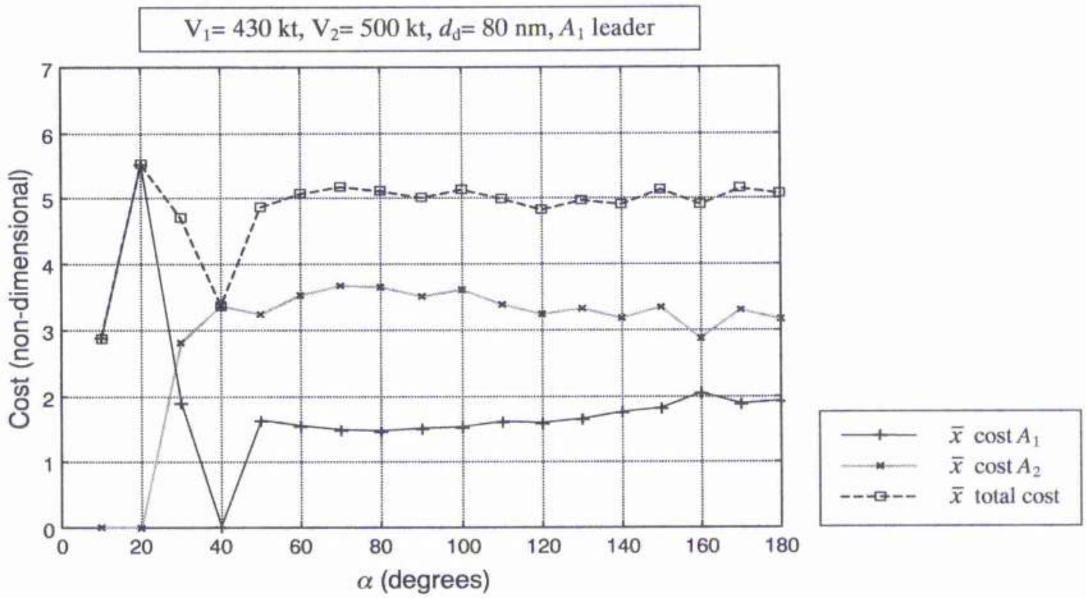
Assuming fixed values for V_1, V_2 and d_d, the application of the co-operation mechanism in each of the 18 scenarios that result from successively assigning to α the values 10°, 20°, 30°,..., 160°, 170°, 180° has been repeatedly simulated 50 times assuming that A_1 acts as the leader and other 50 times assuming that A_2 acts as the leader. In each simulation run, the response latency of the leader's flight crew has been selected at random from the interval [40 s, 60 s], according to the probability density function in Figure 3.6. For each of the values of α considered, each of the two corresponding series of 50 simulation runs results in a random sample of size 50 from each of the four random variables c_1, c_2, s_c and $d_{A1\text{-}A2}$ for the conflict scenario corresponding to that α. Only the samples from c_1, c_2, and s_c are considered in this analysis. The three samples from the variables c_1, c_2, and s_c obtained assuming that A_1 is the leader are drawn according to the conditional probability density functions $f(c_1| E_1)$, $f(c_2| E_1)$ and

$f(s_c| E_1)$, respectively, while the three samples obtained assuming that A_2 is the leader are drawn, respectively, according to the conditional probability density functions $f(c_1| E_2)$, $f(c_2| E_2)$ and $f(s_c| E_2)$. To illustrate the variations in the performance of the co-operation mechanism for the different values of α, the means of the samples obtained, which are considered as point estimates of the means of the corresponding PDFs, are plotted against α. These plots have been generated for four different combinations of values of the parameters V_1, V_2 and d_d.

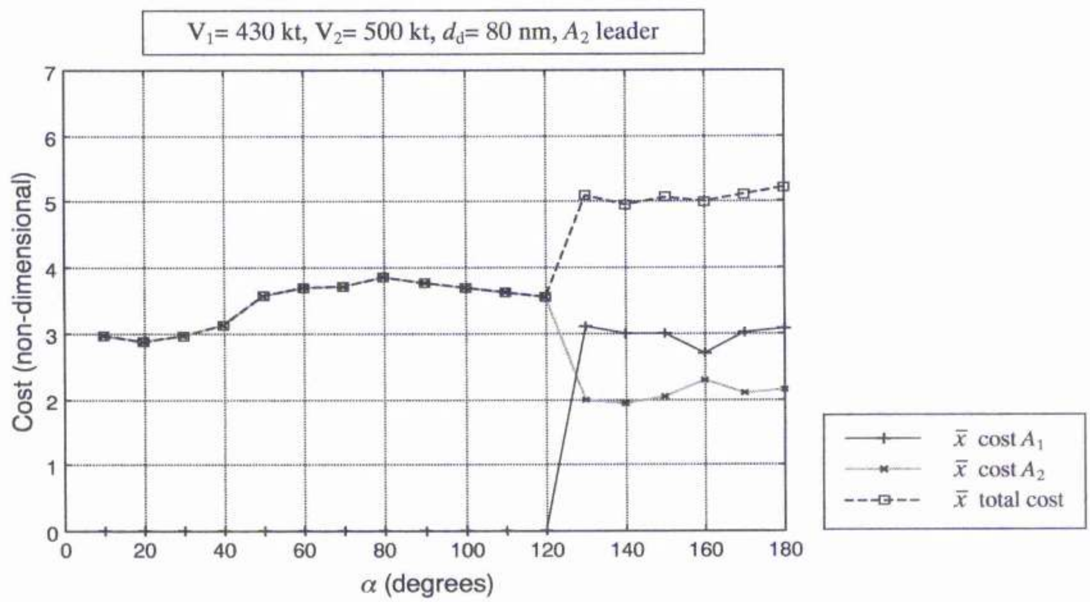
Figure 3.29 shows the results obtained assuming that the values of V_1, V_2 and d_d are the same as in scenarios 1 and 2. Thus, $V_1 = 430$ kt, $V_2 = 500$ kt and $d_d = 80$ nm in the 18 scenarios considered. According to Figure 3.29(a), when A_1 acts as the leader for small values of α, the conflict is generally resolved without one of the two aircraft having to modify its initially intended trajectory. This can be explained by the fact that the corresponding conflict scenarios present larger times between conflict detection and CPA, which allows one of the two aircraft to resolve the conflict by itself without incurring excessive cost.

For values of α greater than 40°, both aircraft modify their intended trajectories to resolve the conflict, thereby sharing the total resolution cost. Figure 3.29(b) shows that when A_2 acts as the leader the conflict is resolved without A_1 having to act for $\alpha \leq 120°$. For $\alpha > 120°$ both aircraft modify their initially intended routes to share the costs of the resolution. Comparing Figures 3.29(a) and 3.29(b) it can be seen that, for $\alpha \leq 120°$, the total resolution cost is lower when A_2 acts as the leader since the conflict is resolved by A_2.

The plots in Figure 3.30 show the results obtained assuming that $V_1 = 430$ kt, $V_2 = 500$ kt and $d_d = 90$ nm. In this case, the aircraft's speeds are the same as in Figure 3.29 but the conflicts are detected earlier as a larger ADS-B range of coverage is assumed. Comparing Figures 3.29 and 3.30, it can be seen how the earlier conflict detection influences the performance of the mechanism for the speeds considered. The total resolution cost is lower with $d_d = 90$ nm than with $d_d = 80$ nm for almost all the values of α in both sequences of actions. When A_1 acts as the leader, the conflict is



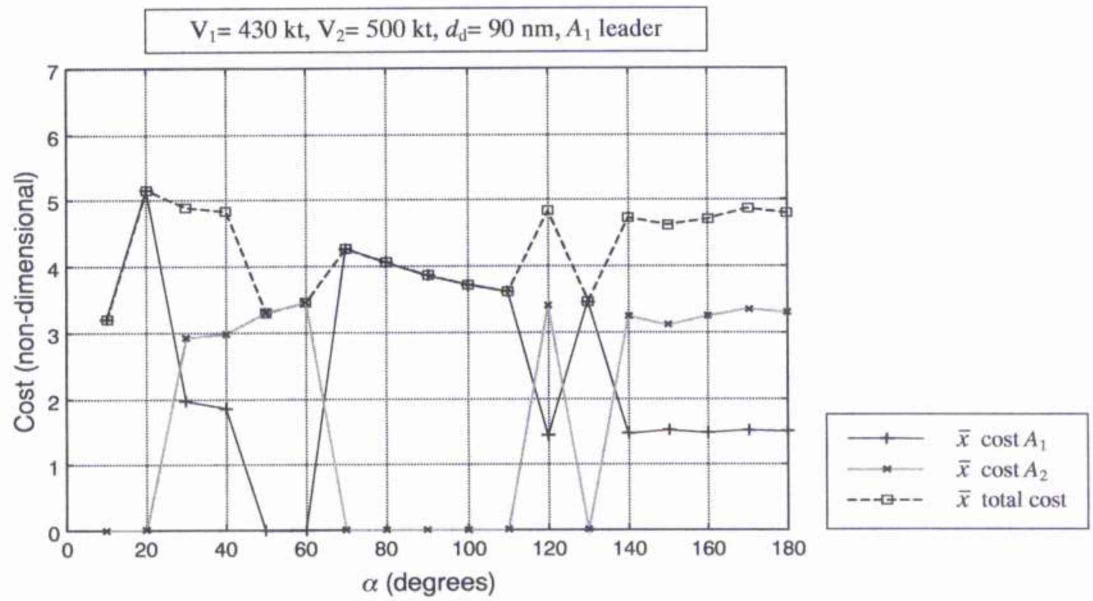
(a) Sample means of the costs assuming that A_1 acts as the leader



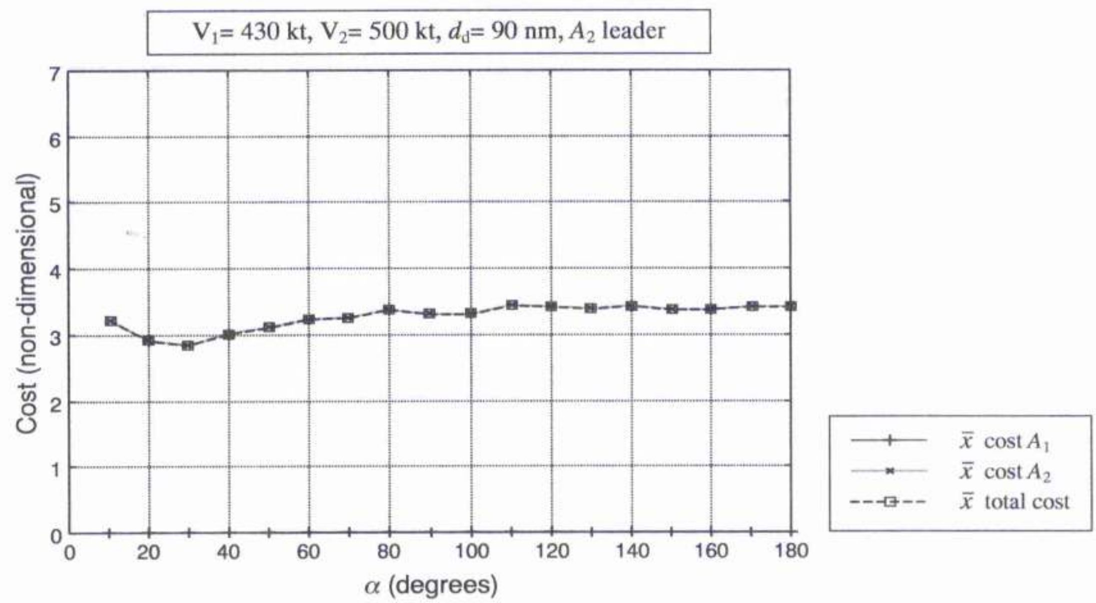
(b) Sample means of the costs assuming that A_2 acts as the leader

Figure 3.29. Performance of the co-operation mechanism for different values of the path crossing angle α given d_d= 80 nm, V_1 =430 kt and V_2= 500 kt. For each value of α, the application of the co-operation mechanism in the resulting conflict scenario is simulated 50 times assuming a sequence of actions; in each simulation run, the leader's flight crew response latency is selected at random from the interval [40 s, 80 s]

resolved without one of the two aircraft having to manoeuvre in more scenarios with d_d = 90 nm than with d_d = 80 nm. When A_2 acts as the leader and d_d = 90 nm, the conflict is resolved without A_1 having to act for all the values of α. In this case, the



(a) Sample means of the costs assuming that A_1 acts as the leader



(b) Sample means of the costs assuming that A_2 acts as the leader

Figure 3.30. Performance of the co-operation mechanism for different values of the path crossing angle α given d_d= 90 nm, V_1 =430 kt and V_2= 500 kt. For each value of α, the application of the co-operation mechanism in the resulting conflict scenario is simulated 50 times assuming a sequence of actions; in each simulation run, the leader's flight crew response latency is selected at random from the interval [40 s, 80 s]

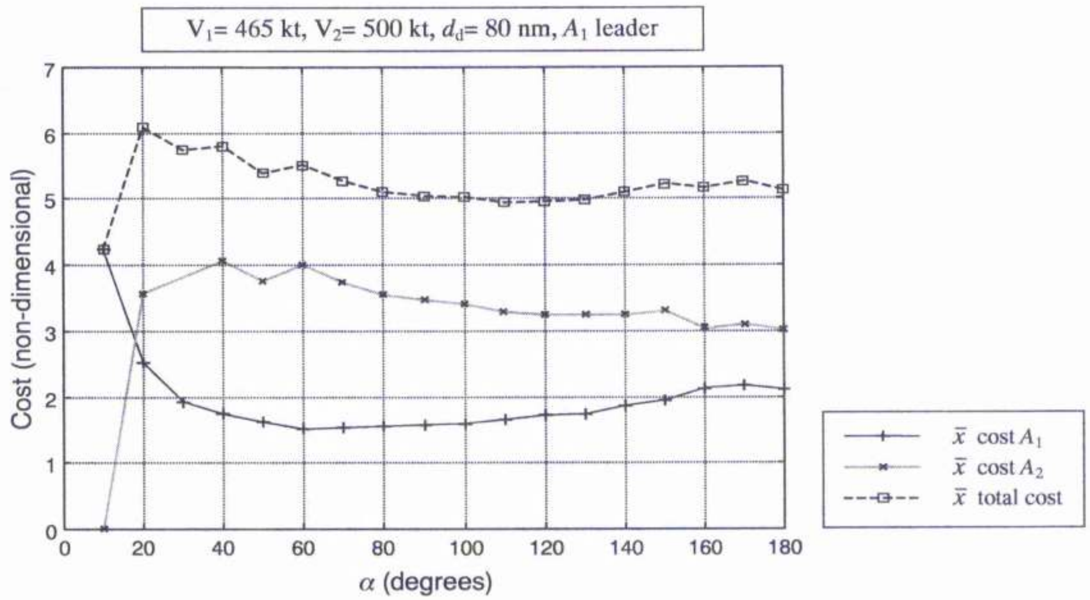
increase in the time between the detection of the conflict and the CPA caused by the larger ADS-B range of coverage enables A_2 to resolve the conflict with a low-cost resolution trajectory regardless of the value of α.

Figure 3.31 illustrates the performance of the mechanism assuming that V_1 = 465 kt, V_2 = 500 kt and d_d = 80 nm. When A_1 acts as the leader, the results obtained are similar to those in Figure 3.29(a). However, it can be noticed that there are more scenarios where the conflict is resolved by only one aircraft with V_1 = 430 kt than with V_1 = 465 kt. When A_2 acts as the leader, the conflict is resolved with A_2's resolution trajectory for $\alpha \le 100°$ with V_1 = 465 kt, while this happens for $\alpha \le 120°$ with V_1 = 430 kt. Comparing Figures 3.29(b) and 3.31(b), it can be noticed that, for the values of α that result in A_1 not having to manoeuvre, A_2's cost is generally higher with V_1 = 465 kt than with V_1 = 430 kt.

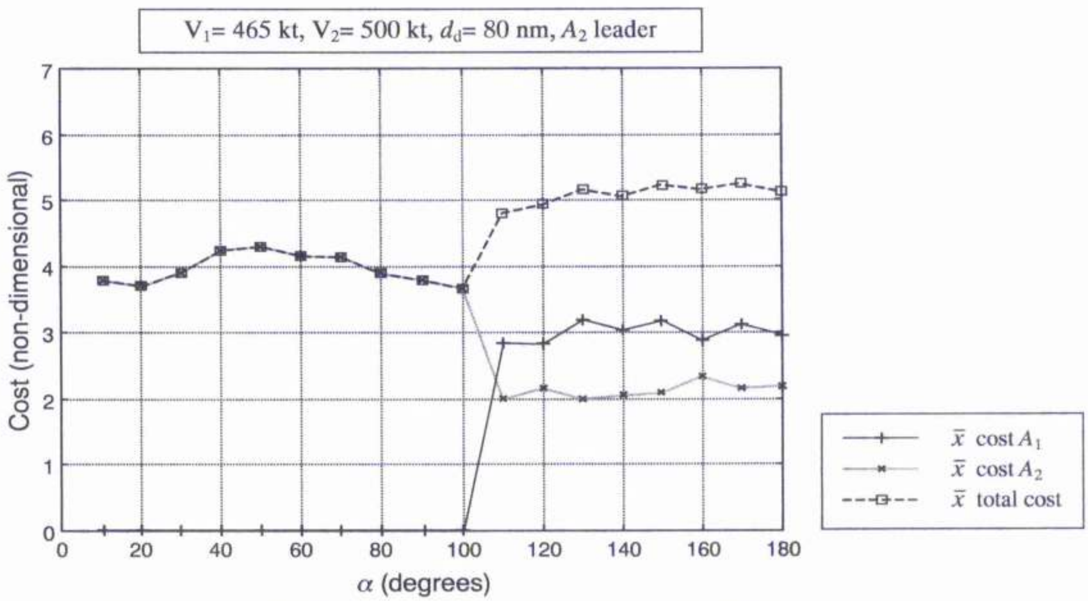
Figure 3.32 shows the results obtained with V_1 = 465 kt, V_2 = 500 kt and d_d = 90 nm. Comparing Figures 3.32(a) and 3.30(a), it can be seen that with V_1 = 465 kt there are more scenarios in which only one of the aircraft has to manoeuvre to solve the conflict. It can also be noticed that the results for the total cost are similar in both figures for nearly all values of α except for $\alpha = 20°$. For $\alpha = 20°$ and V_1 = 465 kt, the algorithm on board A_1 finds a pattern that resolves the conflict without the need for a follower's response, but the iterative improvement process results in a very high cost resolution trajectory compared to the ones obtained for the other values of α. Comparing Figure 3.32(b) with Figure 3.30(b), it can be seen that the results obtained when A_2 acts as the leader are essentially the same with V_1 = 465 kt as with V_1 = 430 kt.

Comparing the four Figures 3.29, 3.30, 3.31 and 3.32, it can be concluded that with the larger value of d_d the conflicts tend to be solved without one of the two aircraft having to manoeuvre. It is also observed that with d_d = 80 nm the number of scenarios in which the two aircraft share the total resolution cost increases as the difference between the aircraft speeds decreases, while the opposite occurs with d_d = 90 nm.

Applying symmetry considerations, the results obtained in this section can be extended to values of α between 180° and 360°. In the next chapter, the performance of the behaviouristic co-operation mechanism will be analysed comparatively with that of the reflective co-operation mechanism proposed there. The analysis will focus on the conflict scenarios investigated in this section and will be concerned with the average



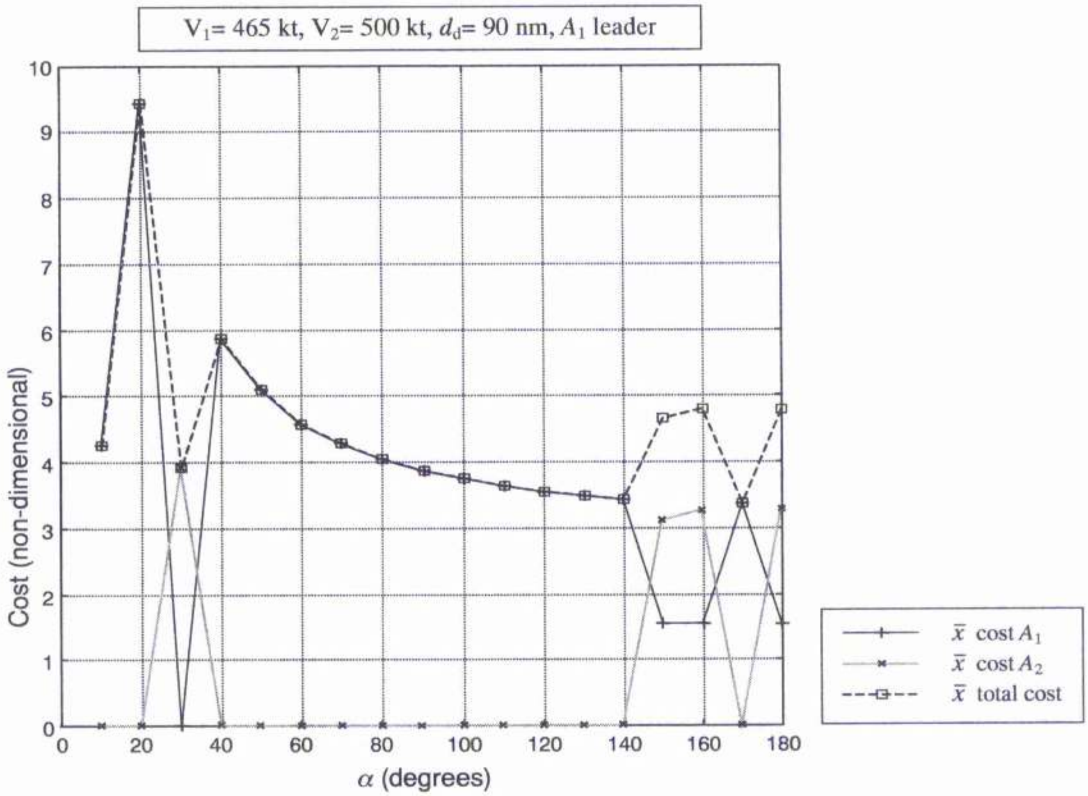
(a) Sample means of the costs assuming that A_1 acts as the leader



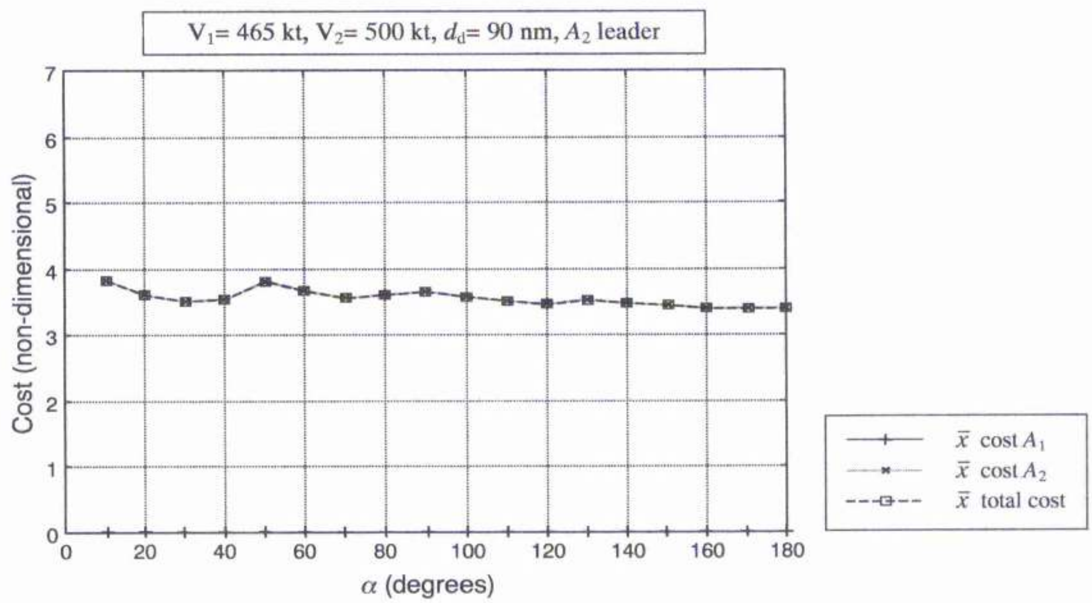
(b) Sample means of the costs assuming that A_2 acts as the leader

Figure 3.31. Performance of the co-operation mechanism for different values of the path crossing angle α given d_d= 80 nm, V_1 =465 kt and V_2= 500 kt. For each value of α, the application of the co-operation mechanism in the resulting conflict scenario is simulated 50 times assuming a sequence of actions; in each simulation run, the leader's flight crew response latency is selected at random from the interval [40 s, 80 s]

performance of the behaviouristic mechanism without assuming which aircraft acts as the leader.



(a) Sample means of the costs assuming that A_1 acts as the leader



(b) Sample means of the costs assuming that A_2 acts as the leader

Figure 3.32. Performance of the co-operation mechanism for different values of the path crossing angle α given d_d= 90 nm, V_1 =465 kt and V_2= 500 kt. For each value of α, the application of the co-operation mechanism in the resulting conflict scenario is simulated 50 times assuming a sequence of actions; in each simulation run, the leader's flight crew response latency is selected at random from the interval [40 s, 80 s]

3.4.6 Two-dimensional conflicts involving three aircraft

This section illustrates the application of the co-operation mechanism in two-dimensional conflict scenarios involving three aircraft. In the scenarios considered, the three aircraft are flying at constant speeds at the same altitude along straight intersecting paths. Two types of conflict scenarios are considered and each of them will be studied separately. In the first type, which is denoted as type I, the three conflicting aircraft are within ADS-B coverage of each other, while in the second type, which is denoted as type II, two of the three conflicting aircraft are outside of the ADS-B range of coverage of each other.

The version of the trajectory-planning algorithm implemented for two-dimensional conflicts between two aircraft has been expanded to make it applicable in conflicts involving three aircraft. To do so, the process for selecting the safe pattern that is subject to the iterative improvement has been modified to accommodate the case of two conflicting followers. Two different selection processes have been incorporated into the algorithm. The aircraft select which selection process to apply depending on the type of conflict scenario.

The MATLAB® integrated computing environment has been used to simulate the application of the co-operation mechanism in the conflict scenarios considered in this section. The simulation process is analogous to the one described in section 3.4.5 for two-dimensional conflicts between two aircraft. Once the flight crew response latencies of the three conflicting aircraft have been specified, the resulting sequence of actions is established and the trajectory-planning algorithm is run for each of the three aircraft. Subsequently, the resulting resolution trajectories are generated and visualised. The simulations have been run using the same version of MATLAB® and the same computing platform as those used for two-dimensional two-aircraft conflicts.

3.4.6.1 Application of the co-operation mechanism in type I conflict scenarios

This section shows how the co-operation mechanism is applied in type I conflict scenarios, which involve three aircraft within ADS-B coverage of one another. The performance of the mechanism in different scenarios of this type is statistically analysed.

The application of the co-operation mechanism in a type I conflict scenario will be explained with an example. The aircraft in scenario 3, which is depicted in Figure 3.33, will be shown to co-operatively resolve the conflicts in which they are involved by applying the co-operation mechanism. It is assumed that the ADS-B range of coverage in this scenario is 80 nm. Three conflicts are expected to occur if the three aircraft maintain their intended routes. As it is shown in Figure 3.33(b), the predicted minimum distances between aircraft A_1 and aircraft A_2, between aircraft A_1 and aircraft A_3 and between aircraft A_2 and aircraft A_3 are smaller than 5 nm.

The initial positions of the aircraft in scenario 3, which are shown in Figure 3.33(a), correspond to the time at which A_2 and A_3 enter each other's area of ADS-B coverage. From that time onwards the three aircraft are within ADS-B coverage of one another. Before that time, A_2 and A_3 were not aware of each other but had detected their respective conflicts with A_1 and were applying the co-operation mechanism to solve them. A_1 had already detected its conflicts with both A_2 and A_3 and was already applying the co-operation mechanism considering the three-aircraft scenario. When A_2 and A_3 detect each other, they inform their respective flight crews and cancel the application of the co-operation mechanism considering only their conflict with A_1. Subsequently, the three aircraft start applying the co-operation mechanism considering the three conflicts. Considering the model of the flight crew response latency adopted (see Figure 3.6), it is assumed that, by the time A_2 and A_3 detect each other, neither of their flight crews has accepted a resolution trajectory for their conflict with A_1.

Each of the three aircraft in scenario 3 considers the initial inter-aircraft distances and the ADS-B range of coverage and concludes that the other two aircraft are also aware of the three conflicts and that are willing to resolve them co-operatively. The co-operation mechanism is applied as follows. Once the three aircraft have detected the three conflicts, each of them runs the trajectory-planning algorithm assuming that it will act as the leader. Firstly, the algorithm searches for a pattern that will serve as input to the iterative improvement The criteria used to select this pattern have been adapted to the

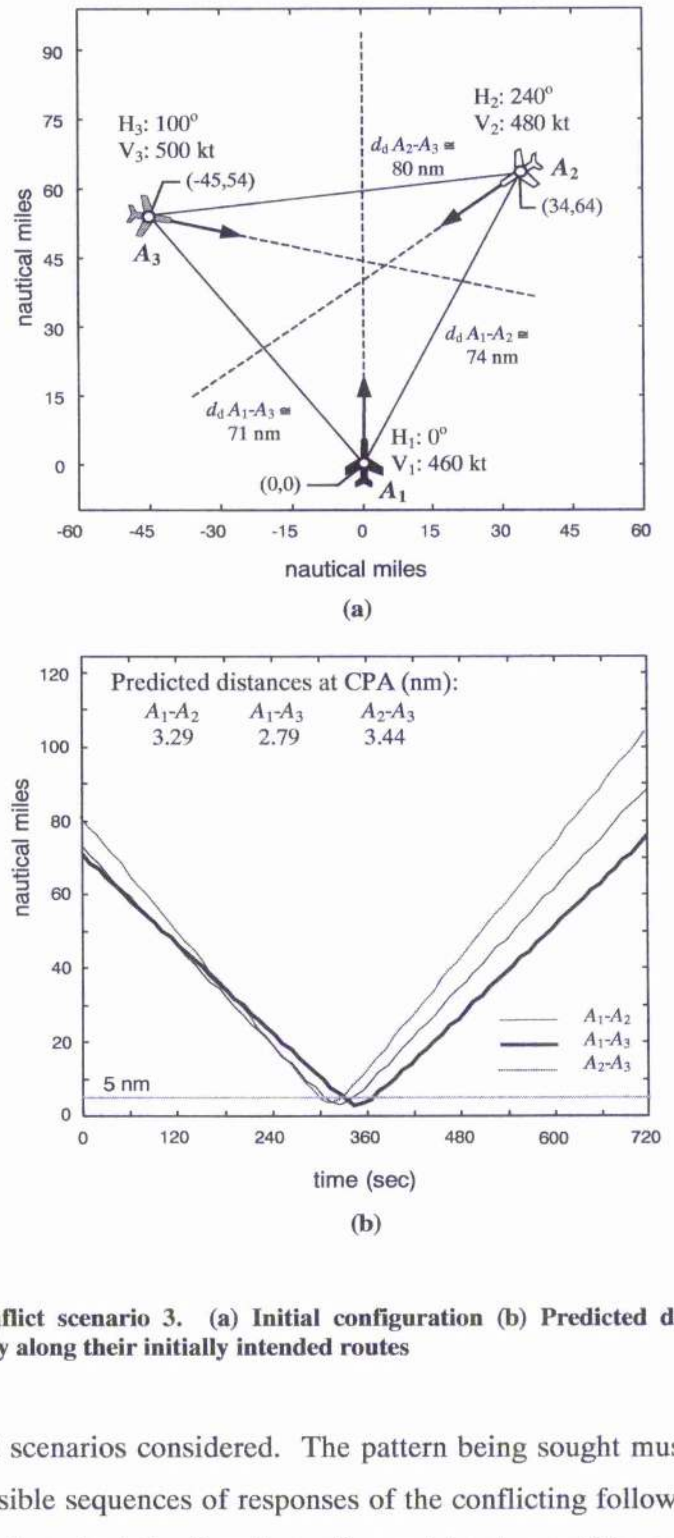


Figure 3.33. Conflict scenario 3. (a) Initial configuration (b) Predicted distances between the aircraft as they fly along their initially intended routes

type of conflict scenarios considered. The pattern being sought must be a safe pattern for the two possible sequences of responses of the conflicting followers. In addition, it is required that the selected safe pattern allows at least one of the conflicting followers to respond with pattern 1, regardless of the sequence of responses. This requirement is intended to facilitate the prediction of the conflicting followers' responses. If none of the safe patterns satisfies this requirement, then the algorithm searches for a safe pattern that allows at least one of the two followers to respond with pattern 1 for one of the two sequences while resulting in the highest possible number of conflict-free combinations of patterns available to the conflicting followers for the other sequence. Finally, if none of the safe patterns satisfies this requirement, then the criterion used in the generic trajectory-planning algorithm is applied. This criterion requires the selected safe pattern to allow for the highest possible number of conflict-free combinations of patterns for the followers assuming they respond following the worst-case sequence (see section 3.3).

Once a safe pattern has been chosen, the iterative improvement process is initiated considering the chosen pattern as the initial candidate resolution trajectory. When the iterative improvement process concludes, the resulting resolution trajectory is presented to the flight crew as the proposed resolution trajectory. For this type of conflict scenario, the CPU time limit for the iterative improvement process has been set to 30 seconds, three times longer than for two-aircraft conflicts. This is justified by the fact that to check whether a combination of trajectories is conflict-free in a three-aircraft conflict the trajectory-planning algorithm has to perform three times as many computations as in a two-aircraft one. The 30 seconds time span for iterative improvement would result in the resolution trajectory being presented to the flight crew too late for it to be accepted according to the model of the flight crew response latency adopted. Considering the increasing computing power available on board modern aircraft, in a hypothetical airborne implementation of the trajectory-planning algorithm the 30 seconds time span could be shortened so that the model of the flight crew response latency adopted is still valid. Thus, in the simulations presented here, it is assumed that the resolution trajectories produced by the algorithm are presented to the flight crew promptly enough to allow for them to be accepted according to the model of the flight crew response latency adopted.

The aircraft whose flight crew accepts first the proposed trajectory acts as the conflicting leader of the other two, which cancel their proposed resolution trajectories and become its conflicting followers. The conflicting followers' resolution trajectories must not conflict with the resolution trajectory broadcast by their conflicting leader. Each conflicting follower applies the trajectory-planning algorithm assuming that it will act as the leader of the other conflicting follower. They use the criteria described above to select a safe pattern among those that do not conflict with the resolution trajectory announced by their conflicting leader. Since at this stage of the conflict resolution process there is only one conflicting follower, these criteria coincide with the ones used in two-aircraft conflicts. The selected safe pattern serves as the input of the iterative improvement process. The conflicting followers present their crew with the resolution trajectory that results from the iterative improvement process. The conflicting follower whose flight crew accepts first the proposed resolution trajectory acts as the leader of the other conflicting follower, which cancels its proposed trajectory and plan a response to the resolution trajectories of its two conflicting leaders.

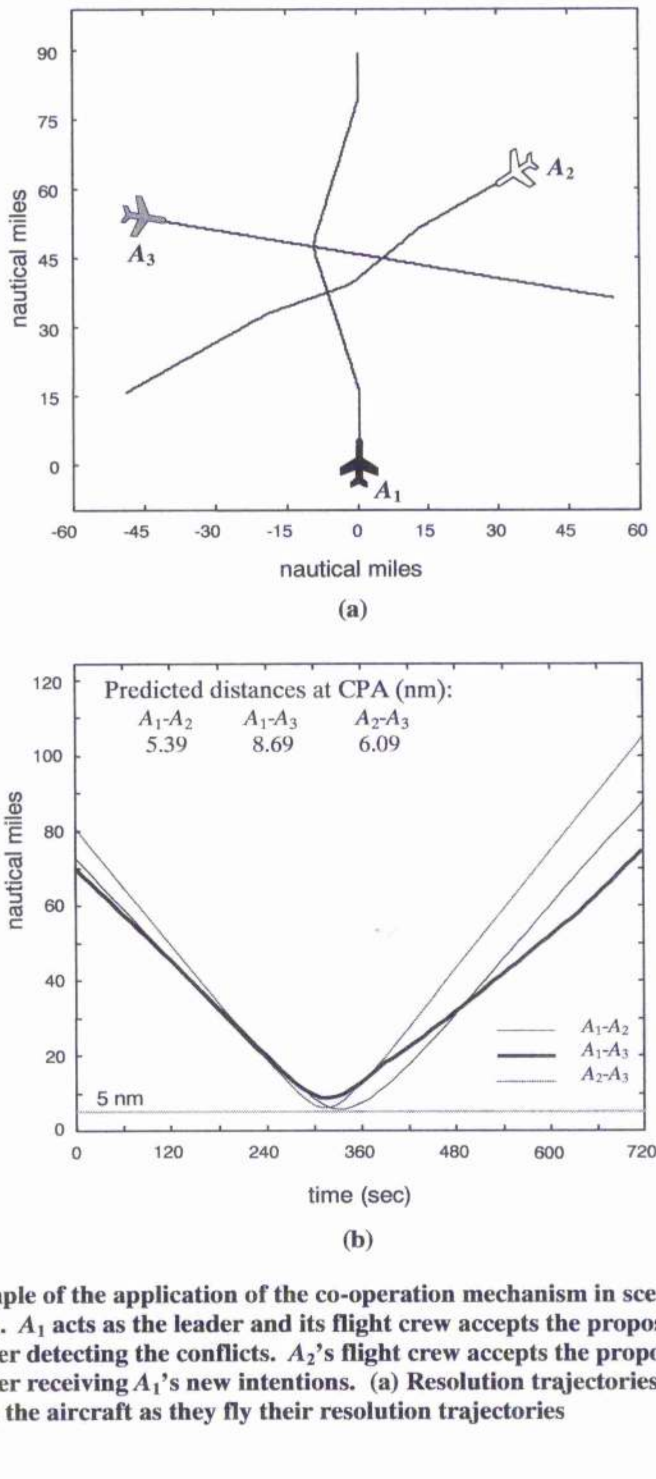
The criteria applied to select the safe pattern that serves as input to the iterative improvement process have been designed to induce the co-operation mechanism to produce resolution strategies in which one of the conflicting aircraft selects pattern 1 as its resolution trajectory. It is assumed that such strategies are preferable to those involving the three conflicting aircraft modifying their initially intended routes. The two manoeuvring aircraft can be sees as acting helpfully towards the non-manoeuvring one. From the point of view of the flight crews, it is assumed that understanding the traffic situation that results from the implementation of a resolution strategy is easier when only two aircraft modify their initial routes. Additionally, the conflicting aircraft that does not have to modify its initially intended route is readily available to perform an emergency resolution manoeuvre, should it be necessary in non-nominal situations such as unforeseen conflicts detected after the resolution strategy has been established.

Considering the foregoing, the application of the co-operation mechanism in scenario 3 has been simulated assuming a sequence of actions of the conflicting aircraft. There are six possible sequences of actions, which are denoted by the following six ordered sequences of the three conflicting aircraft: A_1-A_2-A_3, A_1-A_3-A_2, A_2-A_1-A_3, A_2-A_3-A_1, A_3-A_1-A_2 and A_3-A_2-A_1. These ordered sequences correspond to the six possible permutations of the 3 elements of the set $A_c = \{A_1, A_2, A_3\}$. Once a specific sequence of actions has been assumed, the response latencies of the flight crews are random values sampled according to the PDF in Figure 3.6. The flight crew response latency of the leader is measured from the time at which it detects the three conflicts while the flight crew response latency of a follower is measured from the time at which it receives the resolution trajectory of the previous aircraft in the sequence of actions.

Figures 3.34, 3.36, 3.38, 3.40, 3.42 and 3.44 respectively show an example of the application of the co-operation mechanism in scenario 3 for each of the six possible sequences of actions. Each of these figures depicts the results of simulating the application of the mechanism assuming a sequence of actions and certain values of the response latencies of the flight crews. The values of the parameters defining the resolution trajectories in the figures as well as the costs associated to those trajectories are displayed in Tables 3.9 to 3.14. Additionally, a sequence of snapshots of the positions of the aircraft along the resolution trajectories obtained in each of the examples is shown, respectively, in Figures 3.35, 3.37, 3.39, 3.41, 3.43 and 3.45.

The performance of the co-operation mechanism in scenario 3 has been statistically analysed using the methodology described in section 3.4.5. The application of the mechanism in scenario 3 has been repeatedly simulated 50 times for each of the possible sequences of actions. Assuming a given sequence of actions, the response latencies of the flight crews have been drawn at random from the interval [40 s, 80 s]. These series of simulations result in samples from the random variables modelling the costs of the resolution trajectories, the sum of those costs and the predicted minimum distances between the aircraft. These random variables are denoted, respectively, as c_1, c_2, c_3, s_c, $d_{A1\text{-}A2}$, $d_{A1\text{-}A3}$ and $d_{A2\text{-}A3}$. Each sample has been drawn according to the conditional PDF of the random variable assuming the given sequence of actions. For example, assuming the sequence A_1-A_2-A_3, the sample obtained for the random variable modelling the cost of A1's resolution trajectory, c_1, has been drawn according to the PDF $f(c_1 \mid \text{sequence } A_1\text{-}A_2\text{-}A_3)$. Table 3.15 shows the means of the samples obtained. These sample means can be considered as point estimates of the means of the conditional PDFs according to which they have been drawn.

From Table 3.15 it can be seen that, regardless of the sequence of actions, the resolution strategy resulting from the application of the algorithm generally involves at least one of the conflicting aircraft not having to manoeuvre. The distribution of the total resolution cost among the conflicting aircraft varies depending on the sequence in which the aircraft act. In this scenario, it cannot be assumed that the 6 sequences of actions have the same probability of occurrence. Considering that A_1 detects the conflicts before A_2 and A_3, the sequences in which A_1 acts as the leader can be expected to be more likely to occur than those in which either A_2 or A_3 acts as the leader.



(b)

Figure 3.34. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_1-A_2-A_3. A_1 acts as the leader and its flight crew accepts the proposed resolution trajectory 65 s after detecting the conflicts. A_2's flight crew accepts the proposed resolution trajectory 55 s after receiving A_1's new intentions. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

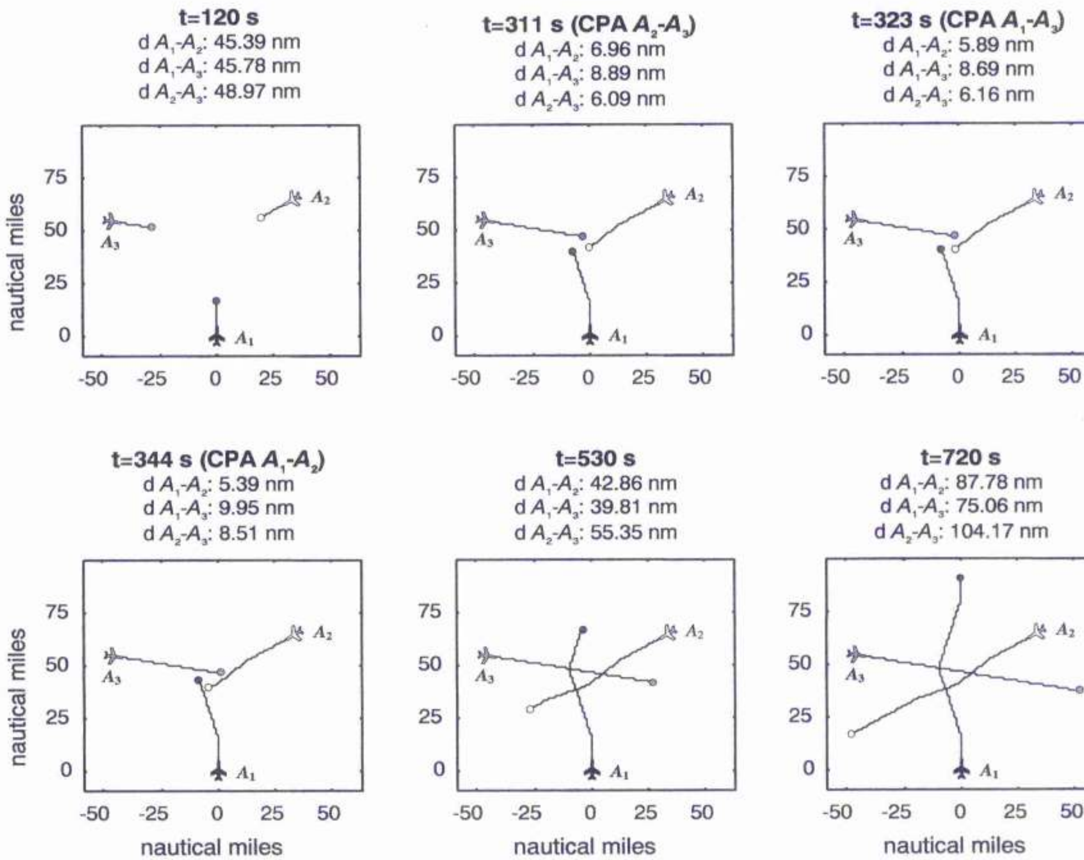
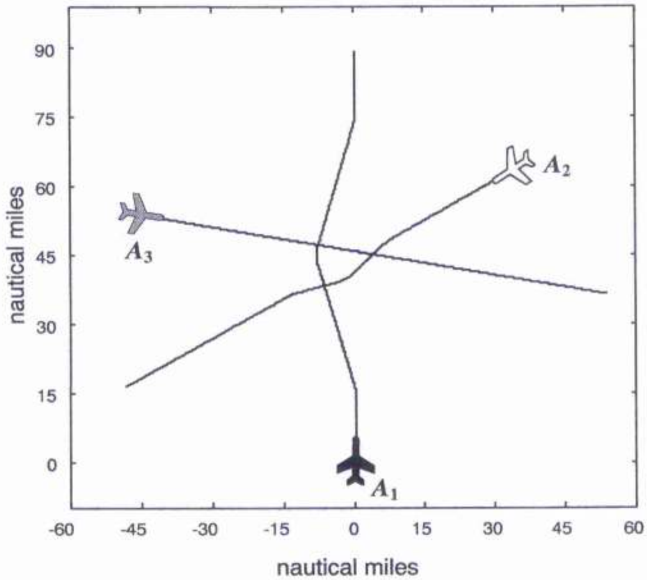


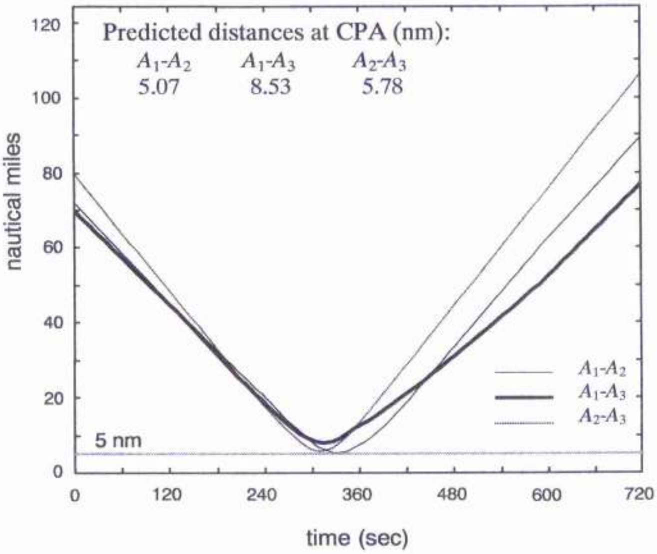
Figure 3.35. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_1-A_2-A_3. Sequence of predicted future positions of the aircraft along the resolution trajectories in Figure 3.34(a)

	γ	t_s	t_e	Cost
A_1	-17°	131 s	641 s	7.10
A_2	-10°	185 s	465 s	3.07
A_3	Initially intended route			0.00
Total conflict resolution cost				10.17

Table 3.9. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_1-A_2-A_3: values of parameters and costs for the resolution trajectories in Figure 3.34(a)



(a)



(b)

Figure 3.36. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_1-A_3-A_2. A_1 acts as the leader and its flight crew accepts the proposed resolution trajectory 70 s after detecting the conflicts. A_3's flight crew accepts the proposed resolution trajectory 45 s after receiving A_1's new intentions. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

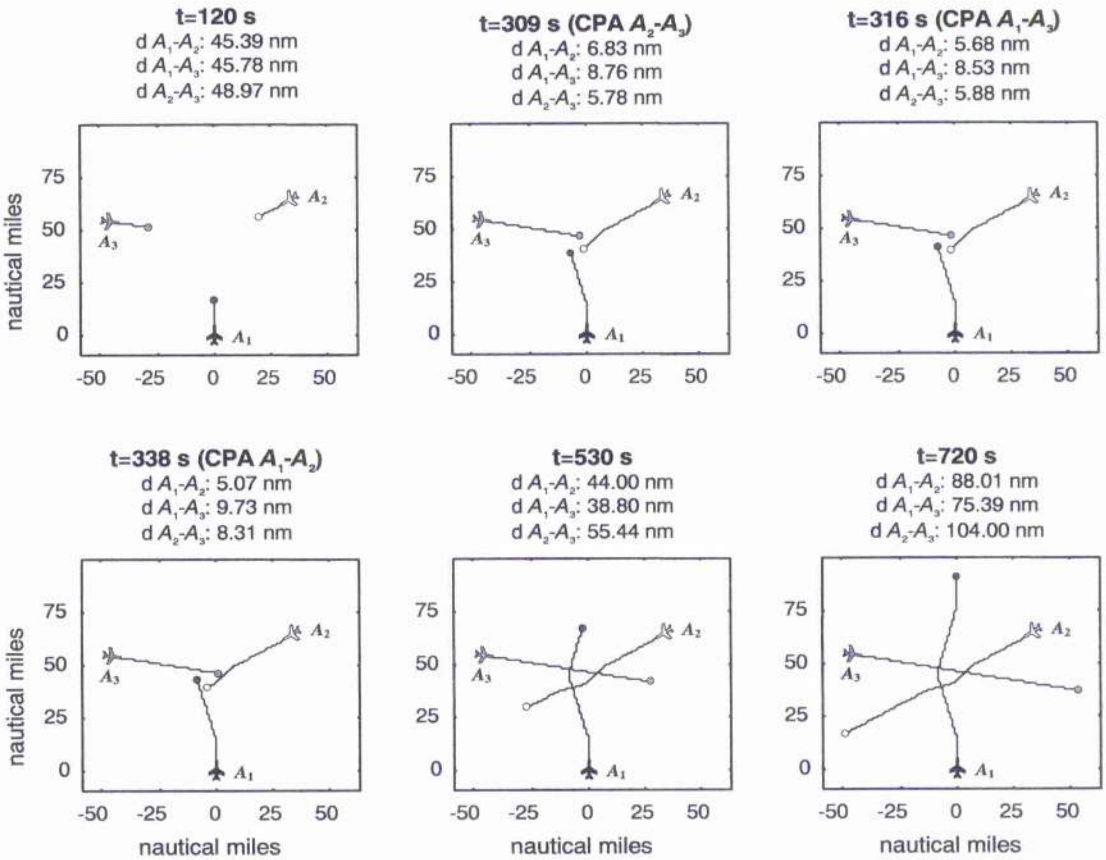
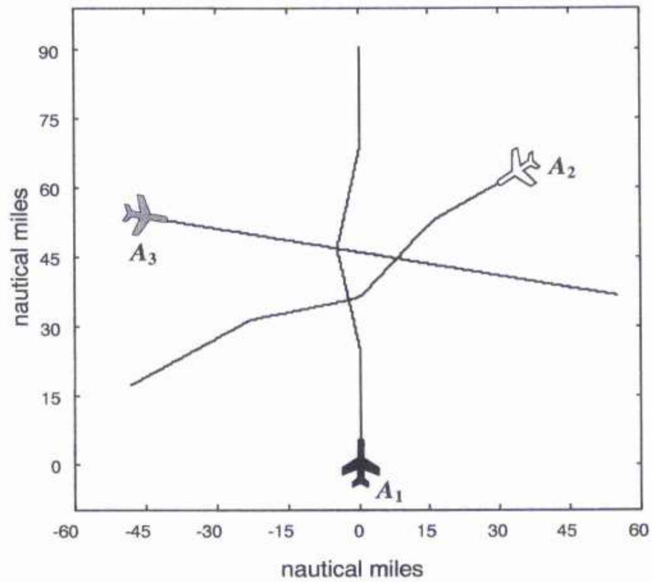


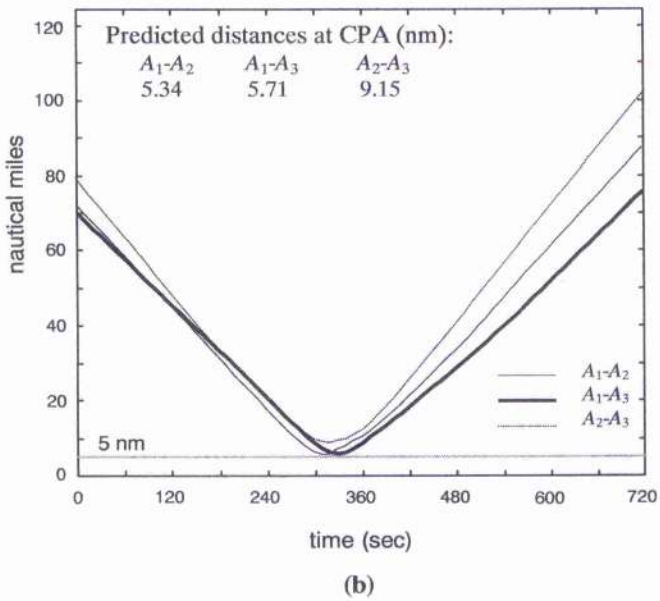
Figure 3.37. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_1-A_3-A_2. Sequence of predicted future positions of the aircraft along the resolution trajectories in Figure 3.36(a)

	γ	t_s	t_e	**Cost**
A_1	-16°	124 s	601 s	6.36
A_2	-15°	235 s	415 s	4.57
A_3	Initially intended route			0.00
Total conflict resolution cost				10.93

Table 3.10. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_1-A_3-A_2: values of parameters and costs for the resolution trajectories in Figure 3.36(a)



(a)



(b)

Figure 3.38. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_2-A_1-A_3. A_2 acts as the leader and its flight crew accepts the proposed resolution trajectory 65 s after detecting the conflicts. A_1's flight crew accepts the proposed resolution trajectory 70 s after receiving A_2's new intentions. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

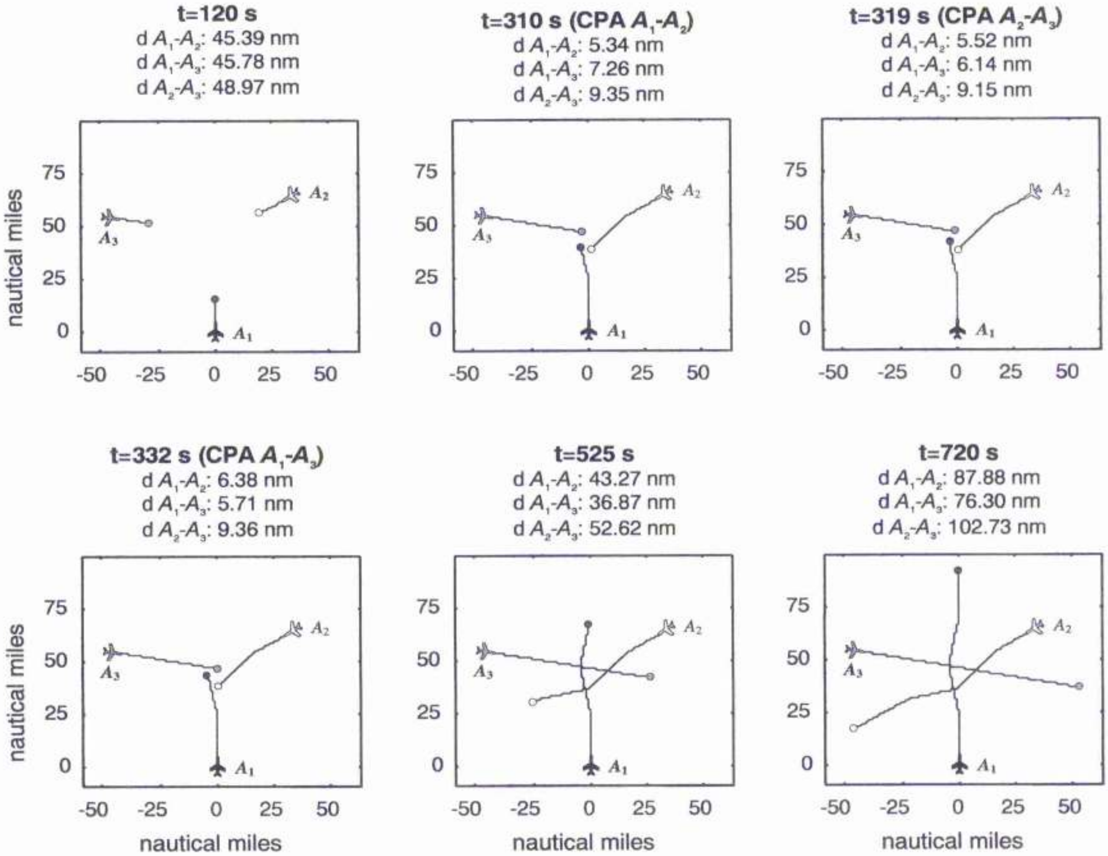


Figure 3.39. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_2-A_1-A_3. Sequence of predicted future positions of the aircraft along their resolution trajectories

	γ	t_s	t_e	Cost
A_1	-12°	192 s	550 s	4.00
A_2	-18°	160 s	506 s	6.76
A_3	Initially intended route			0.00
Total conflict resolution cost				10.76

Table 3.11. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_2-A_1-A_3: values of parameters and costs for the resolution trajectories in Figure 3.38(a)

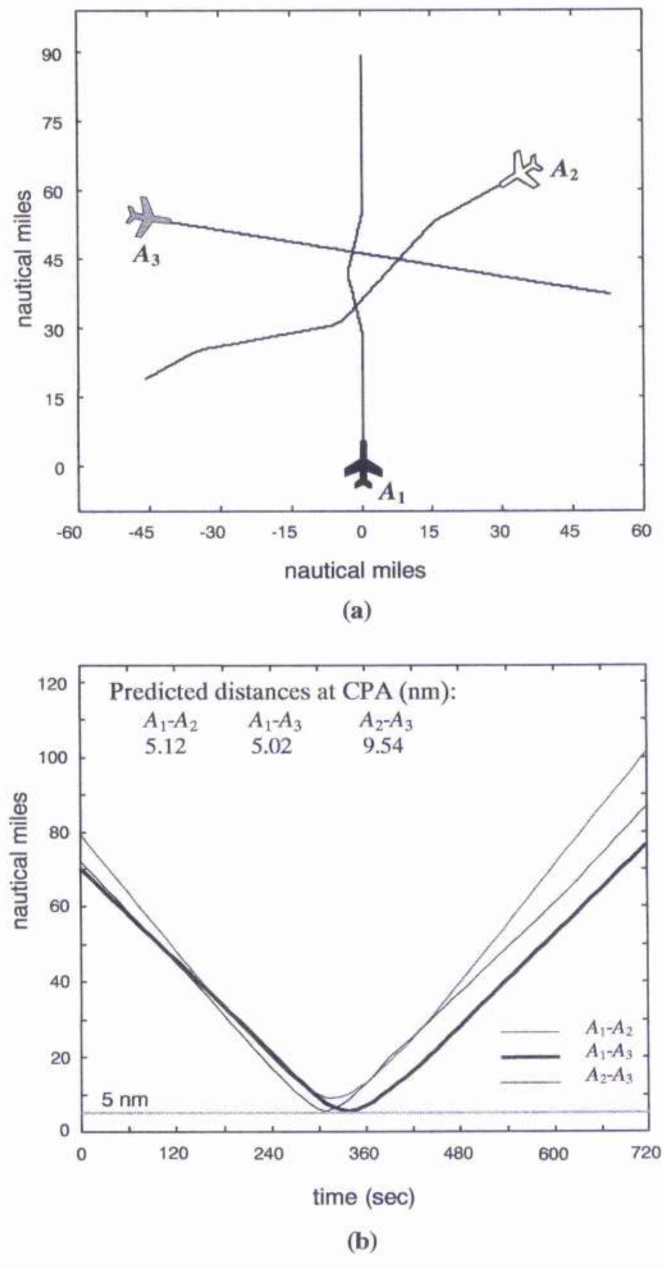


Figure 3.40. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_2-A_3-A_1. A_2 acts as the leader and its flight crew accepts the proposed resolution trajectory 45 s after detecting the conflicts. A_3's flight crew accepts the proposed resolution trajectory 50 s after receiving A_2's new intentions. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

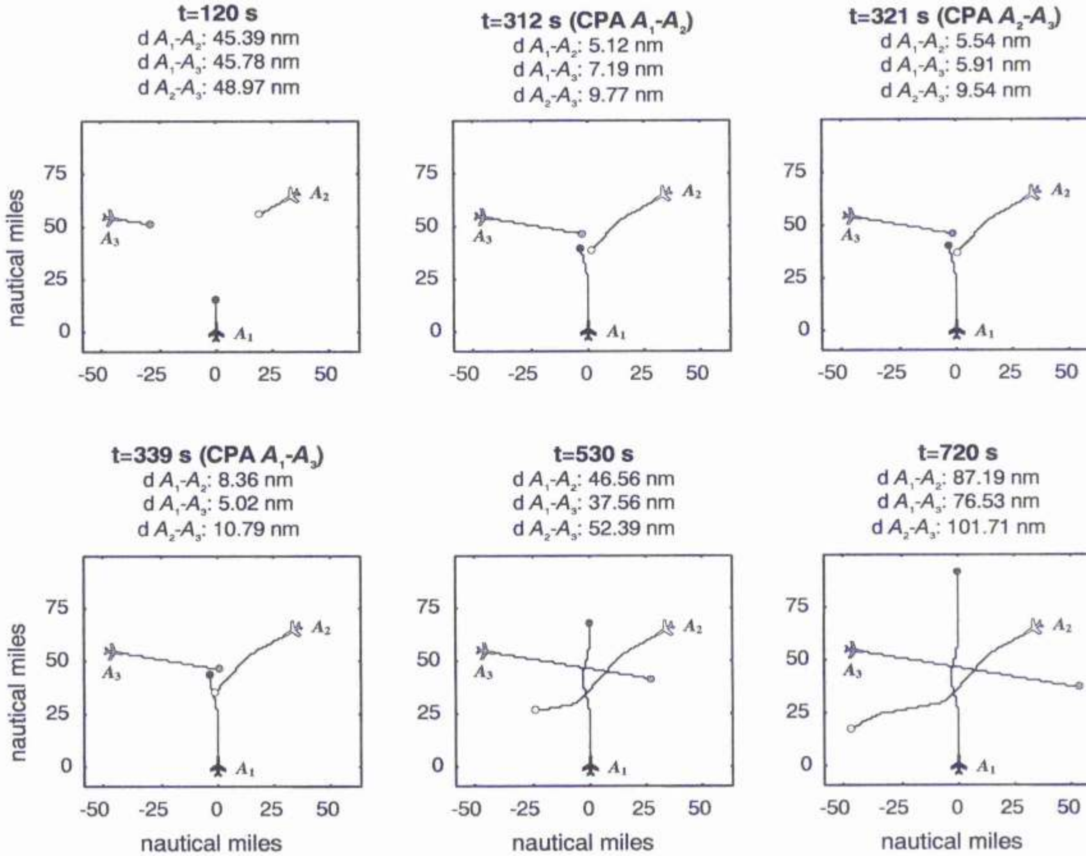
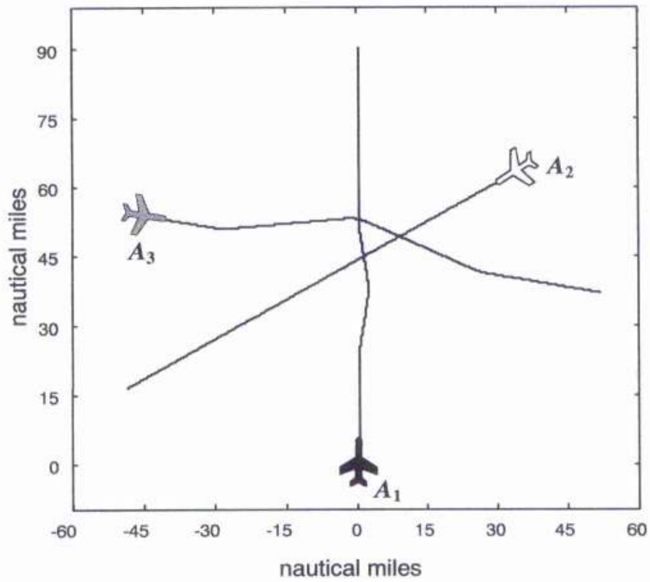


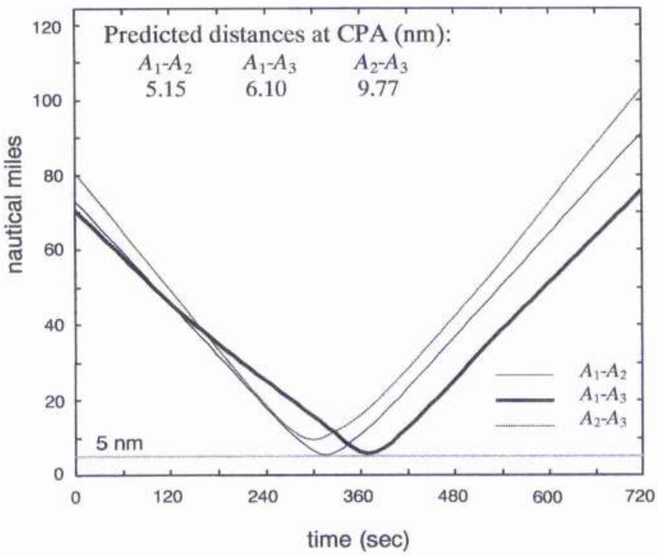
Figure 3.41. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_2-A_3-A_1. Sequence of predicted future positions of the aircraft along the resolution trajectories in Figure 3.40(a)

	γ	t_s	t_e	Cost
A_1	-13°	222 s	433 s	3.94
A_2	-19°	158 s	623 s	8.13
A_3	Initially intended route			0.00
Total conflict resolution cost				12.07

Table 3.12. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_2-A_3-A_1: values of parameters and costs for the resolution trajectories in Figure 3.40(a)



(a)



(b)

Figure 3.42. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_3-A_1-A_2. A_3 acts as the leader and its flight crew accepts the proposed resolution trajectory 60 s after detecting the conflicts. A_1's flight crew accepts the proposed resolution trajectory 65 s after receiving A_3's new intentions. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

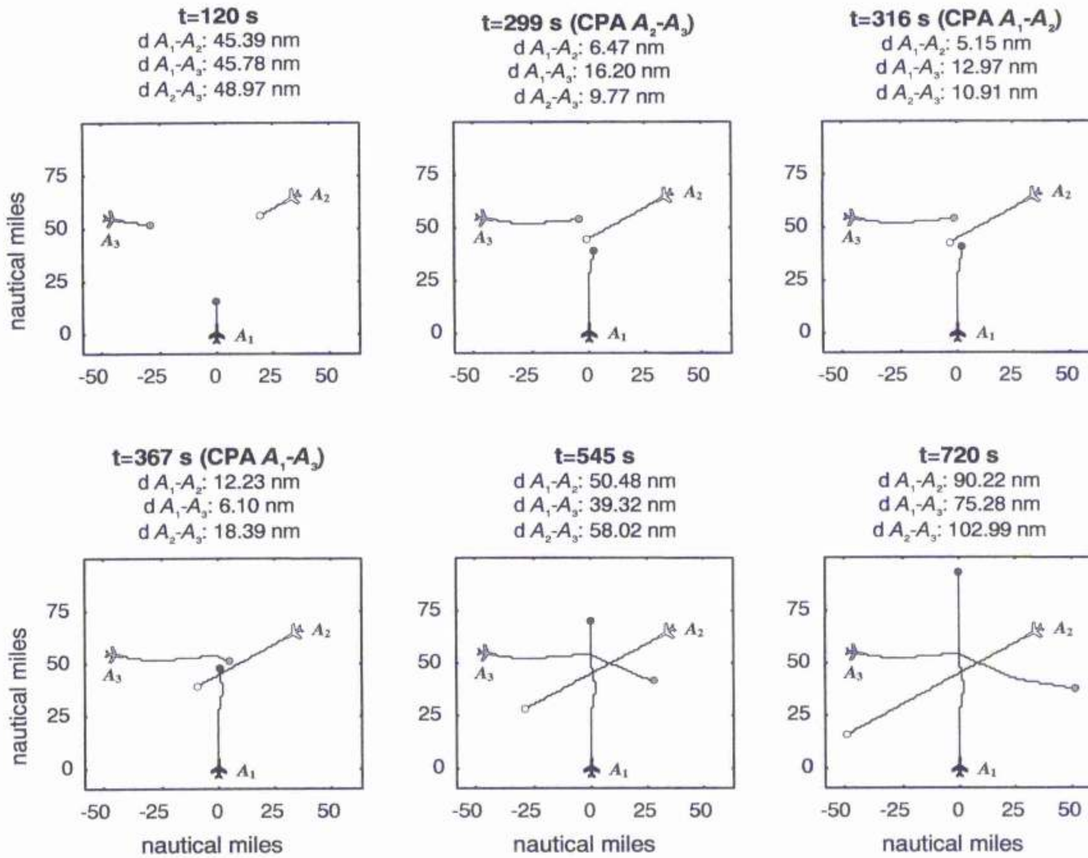


Figure 3.43. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_3-A_1-A_2. Sequence of predicted future positions of the aircraft along the resolution trajectories in Figure 3.42(a)

	γ	t_s	t_e	Cost
A_1	9°	184 s	410 s	2.60
A_2	Initially intended route			0.00
A_3	-15°	120 s	530 s	5.69
Total conflict resolution cost				8.29

Table 3.13. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_3-A_1-A_2: values of parameters and costs for the resolution trajectories in Figure 3.42(a)

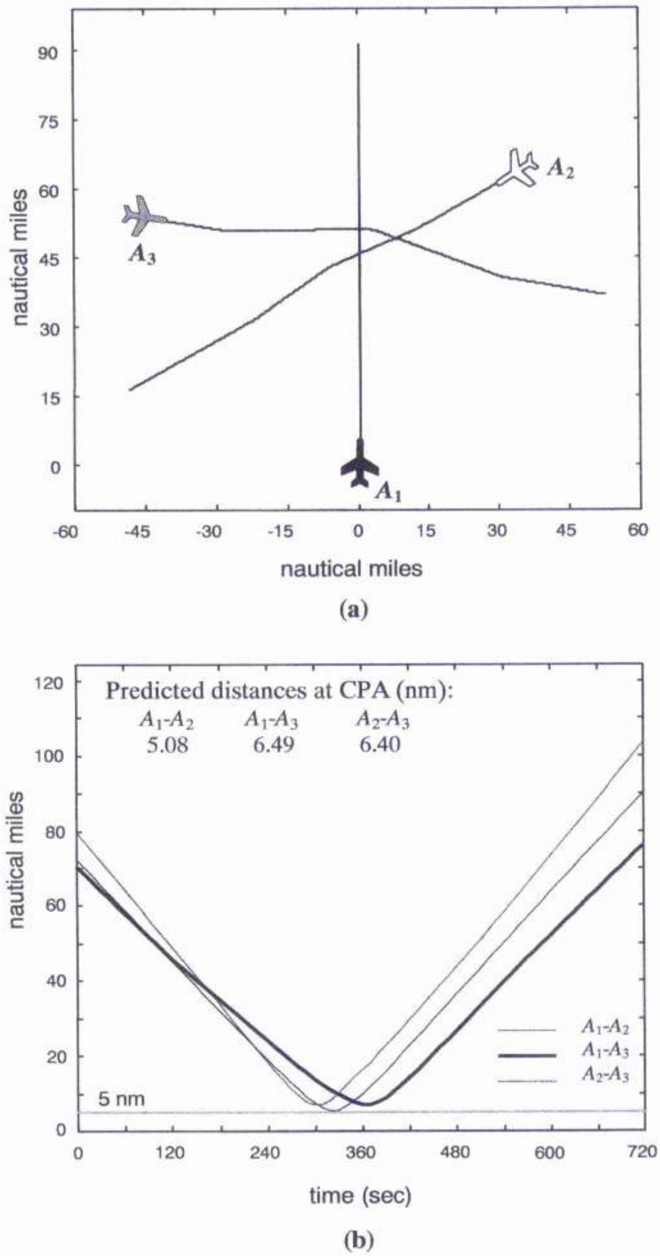


Figure 3.44. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_3-A_2-A_1. A_3 acts as the leader and its flight crew accepts the proposed resolution trajectory 65 s after detecting the conflicts. A_2's flight crew accepts the proposed resolution trajectory 55 s after receiving A_3's new intentions. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

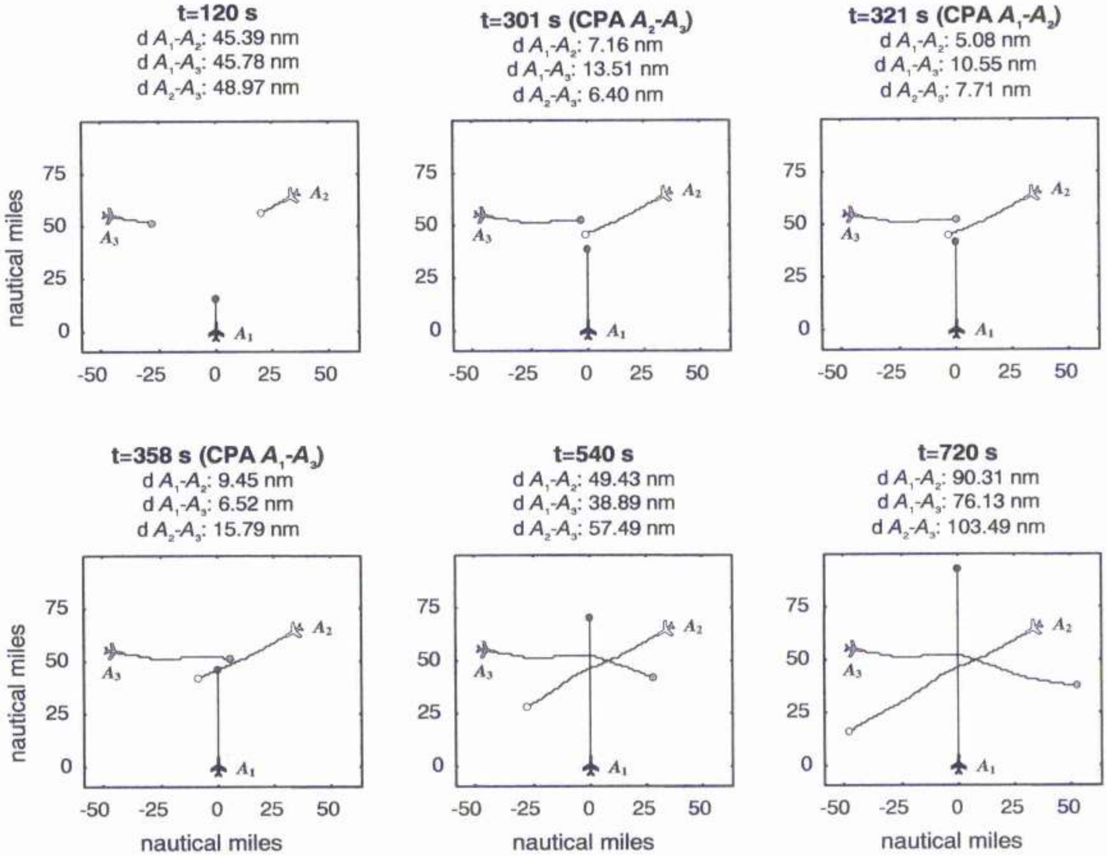


Figure 3.45. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_3-A_2-A_1. Sequence of predicted future positions of the aircraft along the resolution trajectories in Figure 3.44(a)

	γ	t_s	t_e	Cost
A_1	Initially intended route			0.00
A_2	6°	190 s	491 s	1.72
A_3	-11°	127 s	559 s	3.85
Total conflict resolution cost				5.57

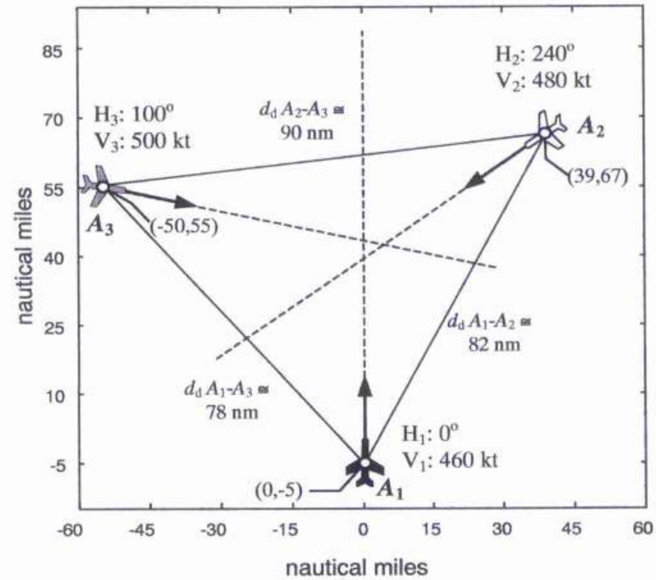
Table 3.14. Example of the application of the co-operation mechanism in scenario 3 assuming the sequence A_3-A_2-A_1: values of parameters and costs for the resolution trajectories in Figure 3.44(a)

		Conflict resolution sequences					
		A_1-A_2-A_3	A_1-A_3-A_2	A_2-A_1-A_3	A_2-A_3-A_1	A_3-A_1-A_2	A_3-A_2-A_1
$\bar{x}$ Cost	A_1	7.54	7.79	3.00	6.05	2.05	1.69
	A_2	3.11	4.76	6.74	6.84	0.00	0.81
	A_3	0.00	0.17	0.31	0.57	5.66	4.10
	Total	10.65	12.72	10.05	13.46	7.71	6.61
$\bar{x}$ CPA distance (nm)	A_1-A_2	5.28	5.35	5.46	5.24	5.19	5.15
	A_1-A_3	8.72	9.00	5.29	5.68	5.59	6.43
	A_2-A_3	5.94	5.53	9.53	9.86	9.71	7.10

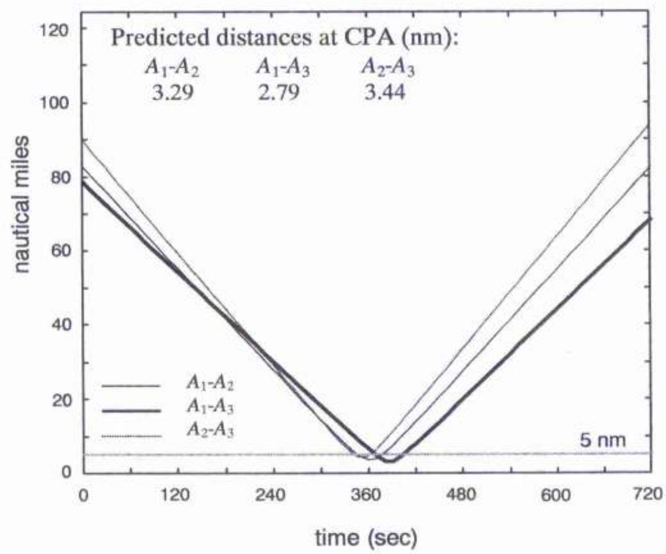
Table 3.15. Simulation of the resolution of the conflicts in scenario 3 for each of the six different possible sequences of action: sample means of the resolution costs and the distances at CPA. Given a sequence, the resolution of the conflicts in the scenario is simulated 50 times; for each simulation run, the flight crew response latencies are selected at random from the interval [40 s, 80 s].

With a view to illustrating the influence of the ADS-B range of coverage on the performance of the co-operation mechanism, the statistical analysis presented above has also been carried for scenario 4, which is shown in Figure 3.40. This scenario represents the conflicting configuration in scenario 3 with an ADS-B range of coverage of 90 nm instead of 80 nm. The aircraft speeds and intended routes are the same in both scenarios and result in the same three conflicts. However, in scenario 4 the aircraft detect the conflicts earlier than in scenario 3. The initial aircraft positions in scenario 4, which are shown in Figure 3.46(a), are the aircraft positions 40 seconds before they reach the initial configuration in scenario 3, which is shown in Figure 3.33(a). The co-ordinates of the aircraft in Figure 3.46(a) have been rounded to the nearest nautical mile for the sake of clarity.

The application of the co-operation mechanism in scenario 4 has been repeatedly simulated 50 times for each of the possible sequences of actions. In each simulation run, the flight latencies of the flight crews are drawn at random from the interval [40 s, 60 s]. Table 3.16 shows the means of the samples obtained from the simulations. Comparing Tables 3.16 and 3.15, it can be concluded that the resolution costs are generally lower with 90 nm ADS-B coverage range than with 80 nm ADS-B coverage range.



(a)



(b)

Figure 3.46. Conflict scenario 4. (a) Initial configuration (b) Predicted distances between the aircraft as they fly along their initially intended routes

As long as each of the three aircraft in a type I conflict scenario is involved in at least one conflict, it can be assumed that the three aircraft are willing to collaborate towards the resolution of all the conflicts in the scenario. Consequently, the co-operation mechanism is applied as it has been shown for scenarios 3 and 4. However, if two of the aircraft in the scenario are in conflict with each other and the third aircraft is not involved in any conflict, then the two conflicting aircraft cannot expect the non-

		Conflict resolution sequences					
		A_1-A_2-A_3	A_1-A_3-A_2	A_2-A_1-A_3	A_2-A_3-A_1	A_3-A_1-A_2	A_3-A_2-A_1
$\bar{x}$ Cost	A_1	5.44	5.58	3.01	4.67	1.62	0.55
	A_2	2.46	3.40	4.95	4.68	0.00	1.17
	A_3	0.00	0.00	0.00	0.25	6.06	5.00
	Total	7.90	8.98	7.95	9.83	7.68	6.72
$\bar{x}$ CPA distance (nm)	A_1-A_2	5.21	5.20	5.35	5.32	5.17	5.17
	A_1-A_3	8.56	8.73	5.66	5.88	6.02	7.44
	A_2-A_3	6.00	5.61	8.96	9.22	11.11	8.17

Table 3.16. Simulation of the resolution of the conflicts in scenario 4 for each of the six different possible sequences of action: sample means of the resolution costs and the distances at CPA. Given a sequence, the resolution of the conflicts in the scenario is simulated 50 times; for each simulation run, the flight crew response latencies are selected at random from the interval [40 s, 80 s].

conflicting one to co-operate with them in the resolution of their conflict. An example will be used to illustrate how the co-operation mechanism is applied in such situations. Consider conflict scenario depicted in Figure 3.47. The ADS-B range of coverage in this scenario is assumed to be 90 nm. As it can be observed in Figure 3.47(b), A_1 is in conflict with A_2 while A_3 is not in conflict with either A_1 or A_2. Thus, A_1 and A_2 have to resolve their conflict considering that A_1 will not co-operate in the resolution process. To do so, they run the trajectory-planning algorithm assuming that A_3 acts as the leader and that its resolution trajectory coincides with its initially intended route. Hence, both A_1 and A_2 consider themselves as conflicting followers of A_3. Each of them assumes that it will act as the leader of the other and plans a resolution trajectory accordingly. Since A_3 is their conflicting leader, their resolution trajectories do not conflict with A_3's initially intended route.

Figure 3.48 shows the results of simulating the application of the co-operation mechanism in scenario 5 assuming that A_1's crew accepts its proposed resolution trajectory before A_2's crew. In this case, A_1 acts as conflicting leader of A_2. A_1's flight crew response latency is assumed to be 55 s. Table 3.17 displays the values of the parameters defining the resolution trajectories in Figure 3.48(a) as well as the costs associated with those trajectories. This example illustrates how the co-operation

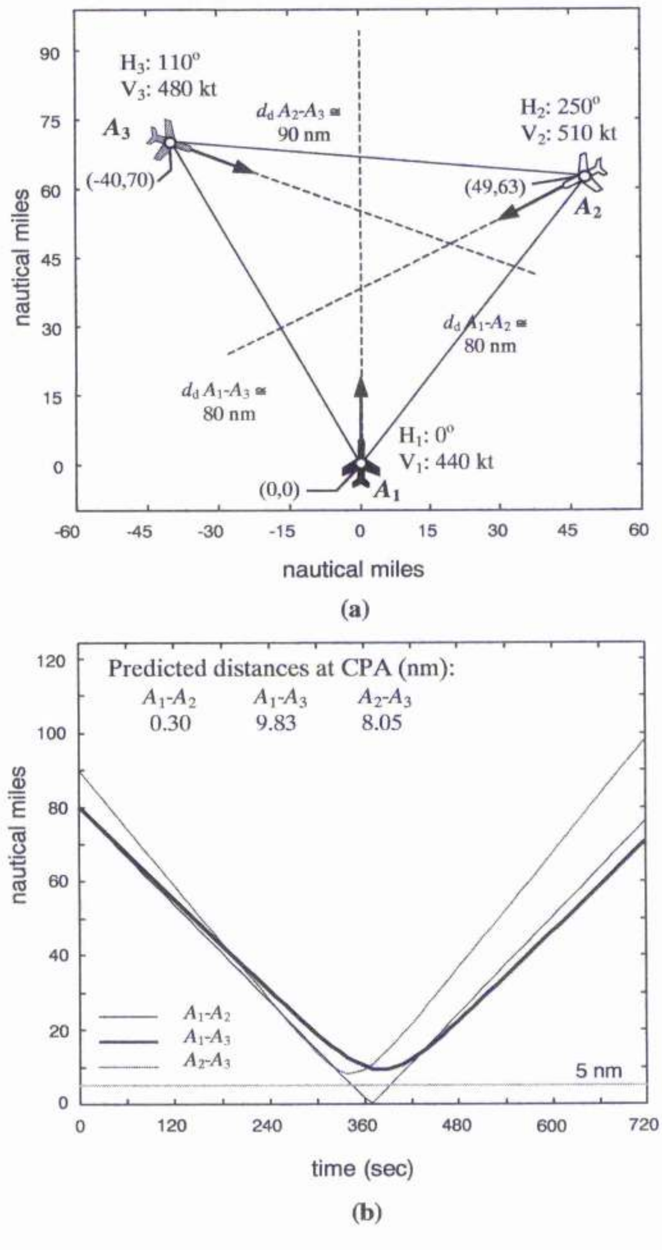
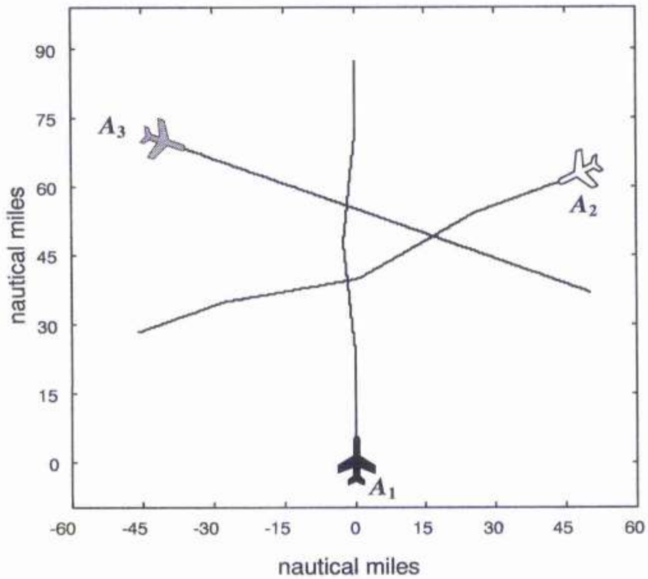
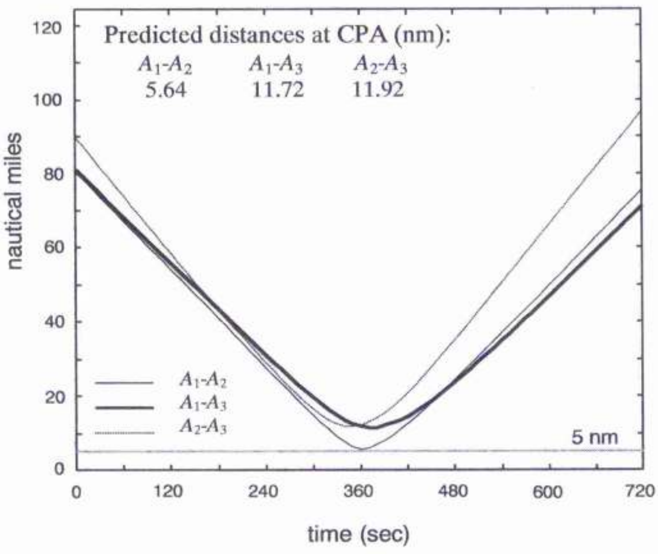


Figure 3.47. Conflict scenario 5. (a) Initial configuration (b) Predicted distances between the aircraft as they fly along their initially intended routes

mechanism enables A_1 and A_2 to share the total cost of the conflict resolution while remaining conflict-free with A_3.



(a)



(b)

Figure 3.48. Simulation of the resolution of the conflicts in scenario 5: A_3 is not involved in any conflict and maintains its initially intended route. Both A_1 and A_2 apply the trajectory–planning algorithm assuming that A_3 is their conflicting leader. A_1 acts first and its flight crew accepts the proposed resolution trajectory 55 s after the detection of the conflicts. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

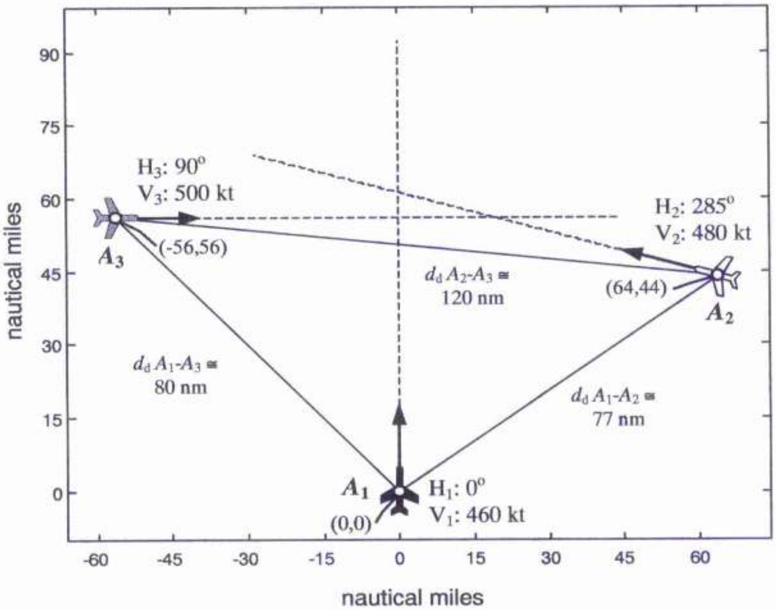
	γ	t_s	t_e	Cost
A_1	-6°	199 s	587 s	1.76
A_2	-10°	175 s	585 s	3.38
A_3	Initially intended route			0.00
Total conflict resolution cost				5.14

Table 3.17. Values of parameters and costs of the resolution trajectories in Figure 3.48(a)

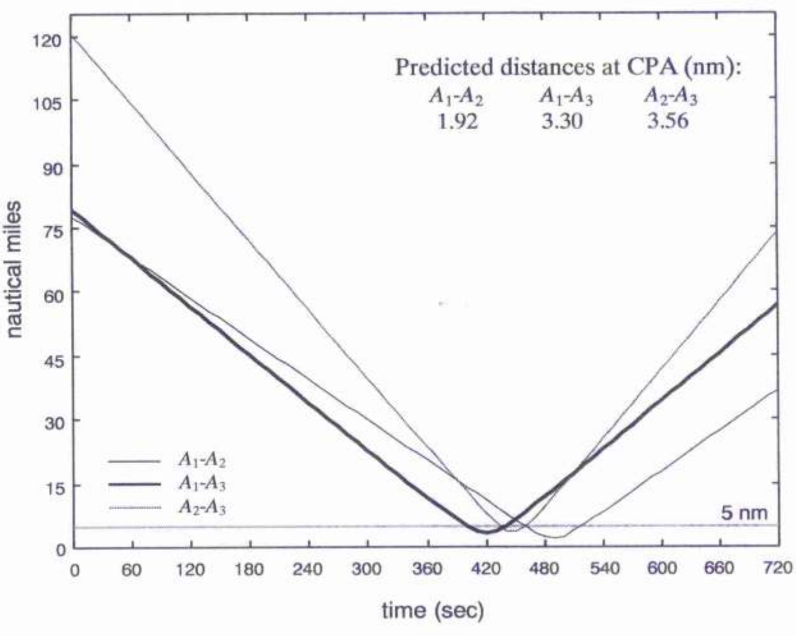
3.4.6.2 Application of the co-operation mechanism in type II conflict scenarios

In this section, the co-operation mechanism is applied in type II conflict scenarios in which each of the three aircraft is in conflict with the other two. Since two of the conflicting aircraft are initially outside of the ADS-B coverage range of each other, only one of the three aircraft is aware of all the conflicts in the scenario. The two aircraft that are outside ADS-B coverage of each other do not detect the conflict between them. The trajectory-planning algorithm has been adapted to such conflict scenarios to enable the aircraft with complete knowledge of the conflicting situation to induce the other two aircraft to unknowingly resolve their conflict. The co-operation mechanism will be shown to provide the means for the two aircraft with incomplete knowledge of the conflicting situation to indirectly collaborate towards the resolution of all the conflicts in the scenario.

The process of applying the co-operation mechanism in the conflict scenarios considered in this section will be explained with an example. Conflict scenario 6, shown in Figure 3.49, is a type II conflict scenario in which each aircraft is in conflict with each of the other two, as it can be seen in Figure 3.49(b). The ADS-B range of coverage in this scenario is assumed to be 80 nm. When the aircraft positions are those displayed in Figure 3.49(a), A_1 detects its conflicts with A_2 and A_3 and both A_2 and A_3 detect their respective conflict with A_1. However, since the distance between A_2 and A_3 is approximately 120 nm, they are outside of the ADS-B range of coverage of each other and therefore unaware of each other's presence and unable to detect the conflict between them. Thus, each of them applies the co-operation mechanism assuming a two-aircraft conflict scenario involving itself and A_1. A_1 is aware of the fact that both A_2



(a)



(b)

Figure 3.49. Conflict scenario 6. (a) Initial configuration (b) Predicted distances between the aircraft as they fly along their initially intended routes

and A_3 apply the co-operation mechanism without considering the conflict between them and that, consequently, their resolution actions may not resolve that conflict. Thus, A_1 runs a version of the trajectory-planning algorithm specifically designed for these situations. This version produces a resolution trajectory that, assuming that A_1

acts as the leader, is highly likely to cause A_2 and A_3 to select resolution trajectories that resolve not only their respective conflict with A_1, but also the conflict between them.

If A_1 acts as the leader, A_2 and A_3 will run the trajectory-planning algorithm to respond to A_1's resolution trajectory considering only their respective conflict with A_1. Their resolution trajectories will therefore be conflict-free with A_1's resolution trajectory but not necessarily conflict-free with each other. The algorithm on board A_1 attempts to produce a resolution trajectory that indirectly causes A_2 and A_3 to produce responses that are conflict-free with each other. However, A_1 cannot predict exactly these responses. Thus, the algorithm on board A_1 produces a resolution trajectory that allows one of the followers to maintain its initially intended route and is simultaneously highly likely to induce the other follower to respond with a resolution trajectory that is conflict-free with that route. Both the safe pattern selection process and the iterative improvement process have been modified to enable A_1 to produce such a resolution trajectory. The resulting version of the algorithm takes advantage of A_1's knowledge of how A_2 and A_3 will respond to its resolution trajectory. A_1 is aware of the fact that if either of its followers is given the chance to maintain its initially intended route, it will certainly do so. Besides, A_1 knows how its followers would run the trajectory-planning algorithm if they had to manoeuvre to respond to A_1's resolution trajectory.

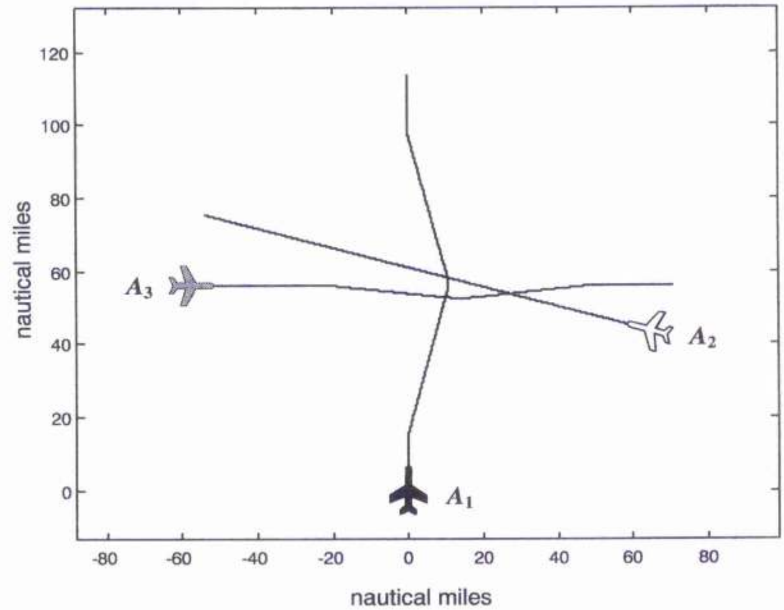
The trajectory-planing algorithm on board A_1 operates as follows. It starts by searching for a conflict resolution pattern that, should it be selected as A_1's resolution trajectory, it will allow one of the two followers to respond by maintaining its initially intended route and the other follower to respond with a pattern that is conflict-free with that route. Once such a pattern has been found it serves as the input for a modified iterative improvement process. This process begins with the generation of a random candidate resolution trajectory that fits the category represented by the selected pattern. If the candidate resolution trajectory allows one of the followers to select pattern 1 as its resolution trajectory, then the algorithm emulates how the other follower would apply the trajectory-planning algorithm to respond to that candidate resolution trajectory. To do so, the algorithm performs the safe pattern selection process and the iterative improvement process that would take place on board that follower as a response to the candidate resolution trajectory. Since A_1 does not know the cost function applied by that follower, the algorithm does not consider the cost of the resolution trajectories generated in the simulated iterative improvement process. Instead, the algorithm counts the number of resolution trajectories in the process that are simultaneously conflict-free with the candidate resolution trajectory of A_1 and the initially intended route of the other follower. This number is denoted as N_s. A_1 stores in memory the candidate resolution trajectory, its associated cost and the value of N_s obtained for that candidate resolution trajectory. Subsequently, a new iteration commences with the generation of another candidate resolution trajectory. If the candidate resolution trajectory generated at the beginning of an iteration does not allow for any of the two conflicting followers to maintain its initially intended route, then that candidate resolution trajectory is discarded and new ones are consecutively generated until a satisfactory one is found.

Successive iterations are performed during a pre-established time span. When this time span expires, A_1's selects the candidate resolution trajectory with the lowest cost among those with the highest value of N_s. The selected candidate resolution trajectory is presented to the crew as A_1's resolution trajectory. A high value of N_s indicates that a high percentage of the random candidate resolution trajectories generated during the iterative improvement process on board the follower that has to manoeuvre are expected to be conflict-free with the initially intended route of the other follower. Thus, A_1's resolution trajectory is guaranteed to be conflict-free with the initially intended route of one of A_1's followers and is simultaneously highly likely to result in the other follower selecting a resolution trajectory that is conflict-free with that route. If the conflict between the two followers is not resolved, the follower that does not modify its initially intend route is available to perform a resolution action once it detects the other follower.

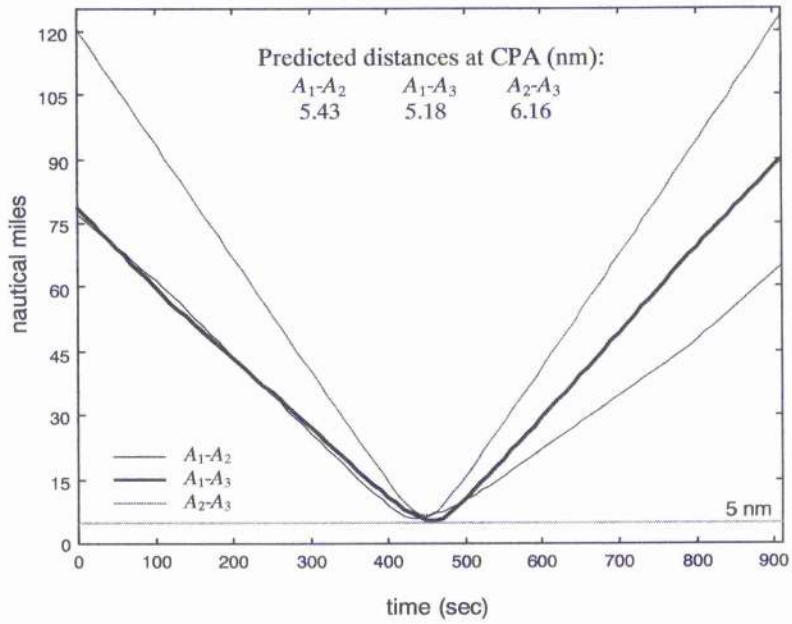
The CPU time span for the iterative improvement process on board A_1 has been set to 100 seconds, to allow the algorithm to produce satisfactory results in the computing platform used to run the simulations. The emulation of the iterative improvement process of the follower in each iteration has been set to take place during a CPU time span of 5 seconds. The 100 seconds CPU time span is incompatible with the model of the flight crew response latency adopted, as the resolution trajectory would be presented to the flight crew more than 80 seconds after the detection of the conflicts (see Figure 3.6). To overcome this contrariety, it is assumed that, were the algorithm to be run on an onboard computer, the 100 seconds CPU time span could be reduced so that the model of the flight crew response model adopted would still be valid. Consequently, in the simulations presented in this section, it is assumed that the resolution trajectories produced by the algorithm are presented to the flight crew promptly enough to allow for them to be accepted according to the model of the flight crew response latency adopted.

Considering the foregoing, the application of the co-operation mechanism in scenario 6 has been simulated assuming that A_1 acts as the leader of both A_2 and A_3 and that its flight crew accepts its proposed resolution trajectory 60 seconds after the detection of the conflicts. A_2's and A_3's flight crews accept their resolution trajectories 45 and 65 seconds after A_1 broadcasts its resolution trajectory, respectively. The results of the simulation are shown in Figures 3.50 and 3.51. Table 3.18 displays the values of the parameters defining the resolution trajectories obtained in the simulation as well as the costs associated with them. In this case, A_1's resolution trajectory is conflict-free with A_2's initially intended route but not with A_3's initially intended route. A_3 has to manoeuvre and its resolution trajectory is not only conflict-free with A_1's resolution trajectory but also with A_2's initially intended route. Thus, A_1 successfully induces A_3 to unknowingly solve its conflict with A_2. If A_3's resolution trajectory conflicted with A_2's initially intended route, A_2 could still respond once it detects its conflict with A_3. In such situation, A_2 would produce a resolution trajectory that is conflict-free with the resolution trajectories of both A_1 and A_3.

The co-operation mechanism is guaranteed to result in the resolution of the three conflicts in the scenario as long as A_1 acts as the leader. However, the mechanism may not result in a conflict-free resolution strategy if either A_2 or A_3 acts as the leader, as A_1 is not able to influence their resolution trajectories so that they resolve their conflict. Hence, an additional feature has been incorporated into the mechanism to ensure that the aircraft-agent with complete knowledge of the conflicting situation in a type II conflict scenario always acts as the leader. This feature operates as follows. As soon as it detects all the conflicts in the scenario, A_1 includes in its ADS-B messages a piece of information indicating that it will act as the leader of the aircraft in conflict with it. When they receive this piece of information, A_2 and A_3 understand that A_1 has better knowledge of the overall conflicting situation than them and decide not to apply the trajectory-planning algorithm until A_1 has announced its resolution trajectory. Thanks to this feature, the co-operation mechanism can be applied successfully in type II conflict scenarios.



(a)



(b)

Figure 3.50. Simulation of the resolution of the conflicts in scenario 6: A_2 and A_3 are outside the ADS-B coverage of each other. A_1 plans a resolution action that aims at inducing A_2 and A_3 to indirectly solve their conflict. A_1 acts first and its flight crew accepts the proposed resolution trajectory 60 s after it detects the conflicts. A_2's flight crew accepts the proposed resolution trajectory 45 s after receiving A_1's new intentions. A_3's flight crew accepts the proposed resolution trajectory 65 s after receiving A_1's new intentions. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

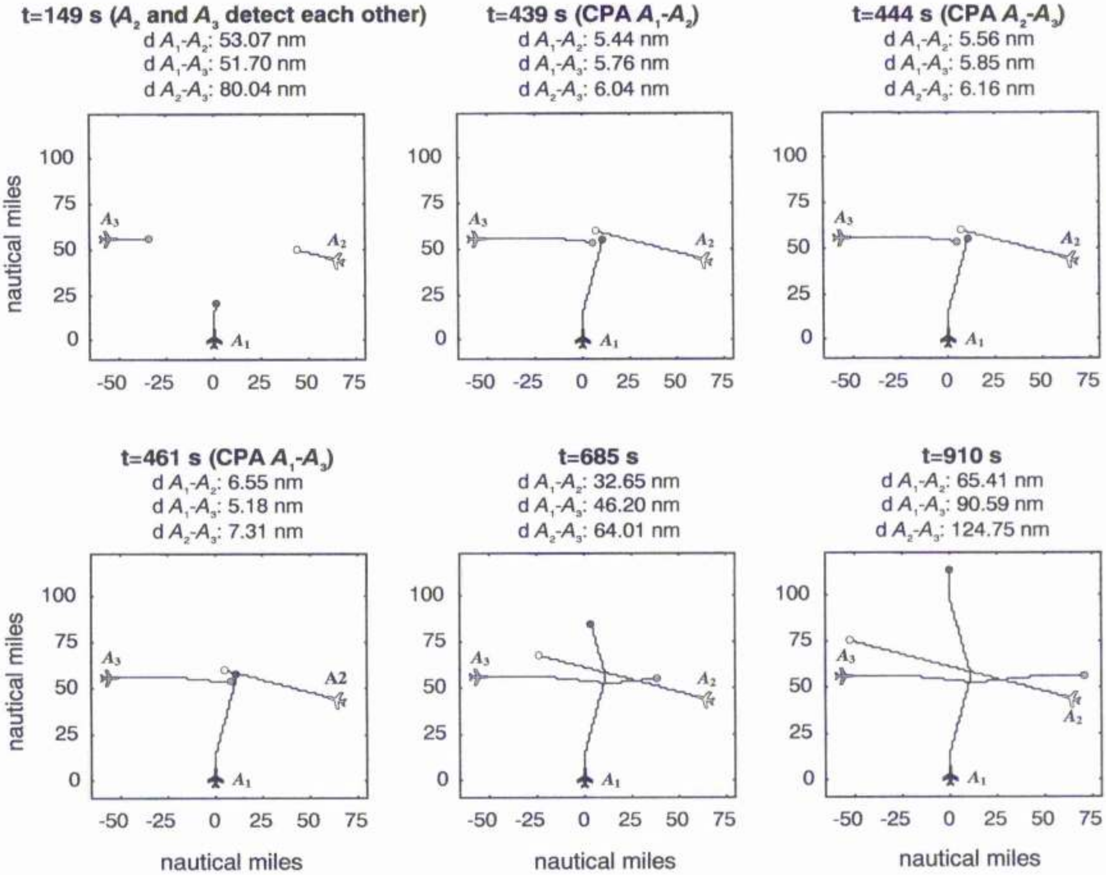


Figure 3.51. Simulation of the resolution of the conflicts in scenario 6: sequence of predicted future positions of the aircraft along their resolution trajectories in Figure 3.50(a)

	γ	t_s	t_e	Cost
A_1	15°	120 s	790 s	6.67
A_2	Initially intended route			0.00
A_3	6°	267 s	746 s	1.86
Total conflict resolution cost				8.53

Table 3.18. Simulation of the resolution of the conflicts in scenario 6: values of parameters and costs of the resolution trajectories in Figure 3.50(a)

In Figure 3.50(b) it can be seen that the distance between A_2 and A_3 is still greater than 80 nm when A_3 accepts its proposed resolution trajectory 125 seconds after it detects its conflict with A_1. Therefore, the resolution strategy is established before A_2 and A_3 become aware of each other's presence. In cases in which the two followers detect the conflict between them before one of them has broadcast its resolution trajectory, then that follower cancels its current resolution trajectory and produces a new one considering the entire conflict scenario. If none of the two followers broadcasts its resolution trajectory before detecting the conflict between them, then both followers cancel their current resolution trajectory and apply the co-operation mechanism to respond to A_1's resolution action considering the entire conflict scenario. In all the scenarios considered in this section the resolution strategy is established before the two followers detect the conflict between them, with a view to illustrating the performance of the co-operation mechanism in such situations.

The performance of the co-operation mechanism in scenario 6 has been statistically analysed using the methodology described in section 3.4.5. The application of the mechanism in scenario 6 has been repeatedly simulated 50 times assuming that A_1's crew accepts the proposed resolution trajectory first in all the simulation runs. In each simulation run, A_1's response latency is drawn at random from the interval [40 s, 80 s]. Since both A_2 and A_3 are assumed to respond to A_1's resolution trajectory before they detect each other, their responses only depend on that resolution trajectory and on A_1's flight crew response latency. Thus, A_2's and A_3's flight crew response latencies do not affect the results of the simulations. The simulations result in samples of size 50 from the random variables modelling the costs of each of the resolution trajectories, the sum of those costs and the minimum predicted distances between each pair of aircraft. Each sample has been drawn according to the corresponding conditional PDF assuming that A_1 acts as the leader. Table 3.19 shows the means of the samples obtained. The three conflicts are solved without A_2 having to manoeuvre in all the simulation runs.

The application of the co-operation mechanism has also been simulated 50 times for conflict scenario 7, which is depicted in Figure 3.52. This scenario represents a conflicting configuration similar to that in scenario 6. The aircraft speeds and headings are the same in both scenarios but in scenario 7 the ADS-B coverage range is assumed to be of 90 nm instead of 80 nm. Table 3.20 shows the means of the samples obtained for conflict scenario 7.

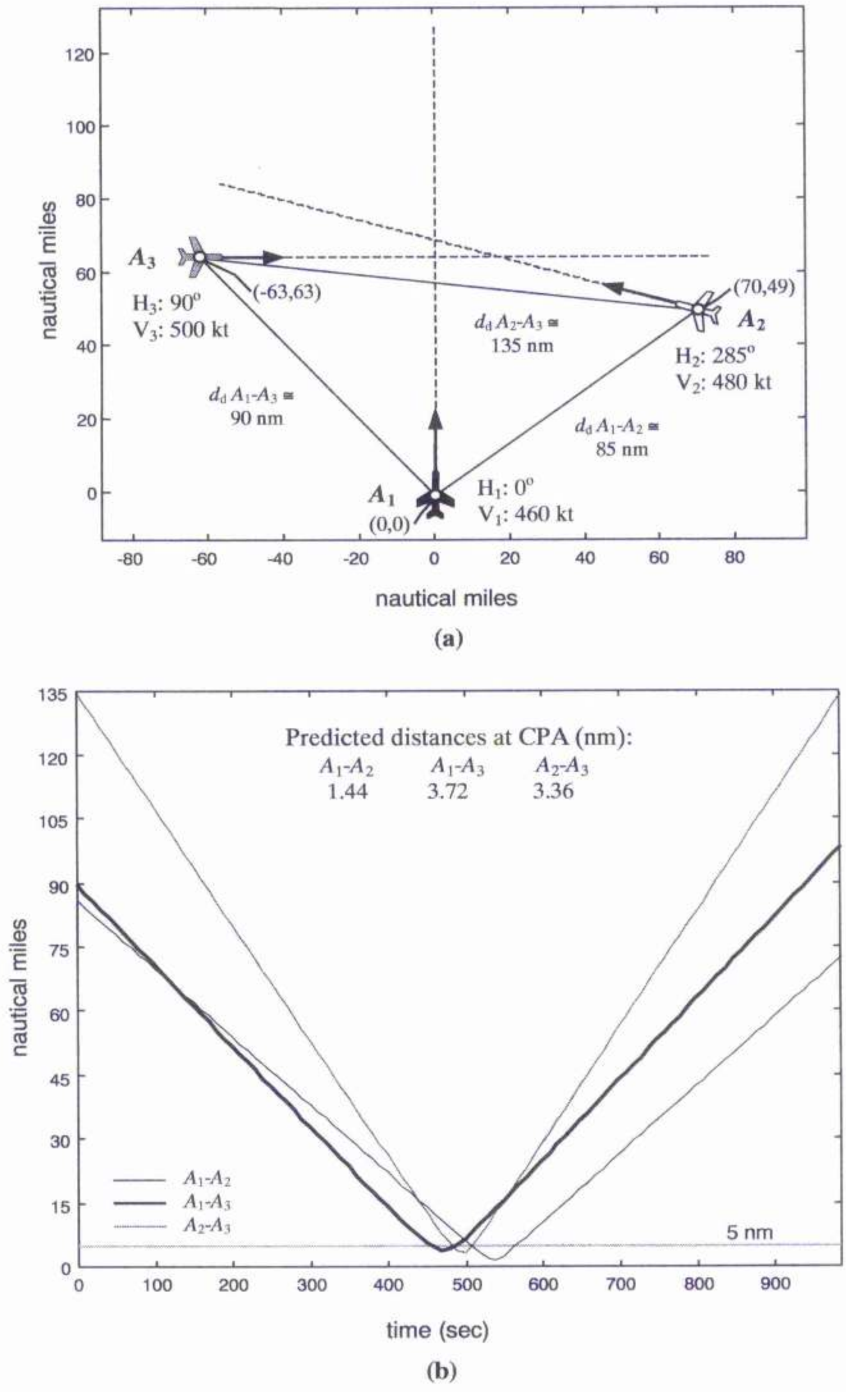


Figure 3.52. Conflict scenario 7. (a) Initial configuration (b) Predicted distances between the aircraft as they fly along their initially intended routes

$\bar{x}$ Cost	A_1	6.69
	A_2	0.00
	A_3	1.74
	Total	8.42
$\bar{x}$ CPA distance (nm)	A_1-A_2	5.39
	A_1-A_3	5.31
	A_2-A_3	6.46

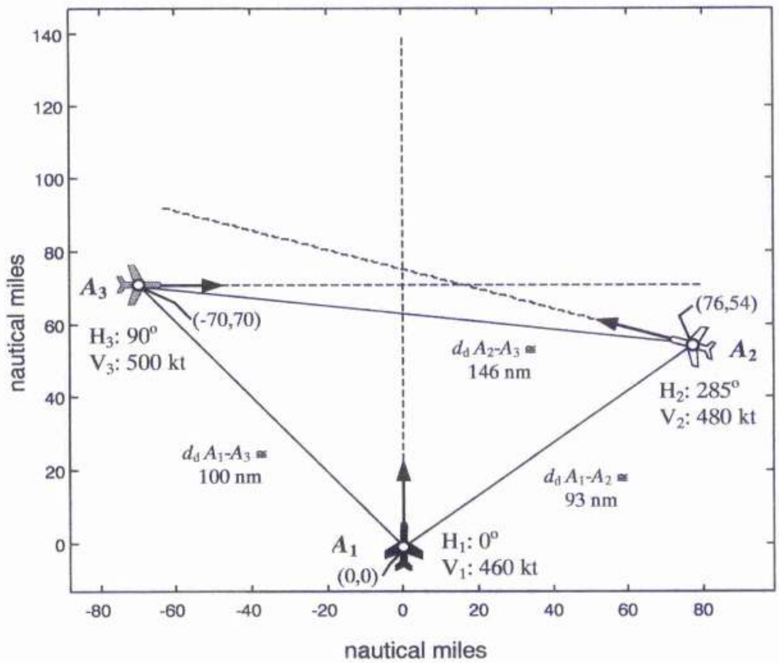
Table 3.19. Sample means of the resolution costs and the distances at CPA obtained from simulating 50 times the resolution of the conflicts in scenario 6 assuming that A_1 acts as the leader and both A_2 and A_3 accept their respective proposed resolution actions before they detect each other. In each simulation run, A_1's flight crew response latency is selected at random from the interval [40 s, 80 s]

$\bar{x}$ Cost	A_1	6.20
	A_2	0.00
	A_3	1.92
	Total	8.13
$\bar{x}$ CPA distance (nm)	A_1-A_2	5.68
	A_1-A_3	5.32
	A_2-A_3	6.74

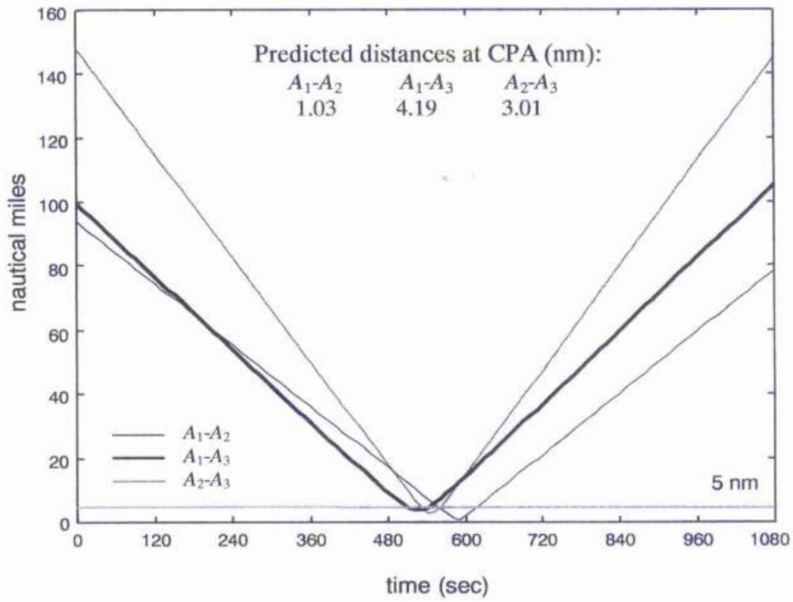
Table 3.20. Sample means of the resolution costs and the distances at CPA obtained from simulating 50 times the resolution of the conflicts in scenario 7 assuming that A_1 acts as the leader and both A_2 and A_3 accept their respective proposed resolution actions before they detect each other. In each simulation run, A_1's flight crew response latency is selected at random from the interval [40 s, 80 s]

Again, the three conflicts are resolved without A_2 having to manoeuvre in all the simulation runs. As it can be seen comparing Tables 3.20 and 3.19, the sample mean of the total resolution cost is slightly lower in scenario 7 than in scenario 6. This can be justified by the fact that the ADS-B coverage range is longer in the former than in the latter.

Finally, the application of the co-operation mechanism has been simulated for conflict scenario 8, depicted in Figure 3.53. This scenario represents a conflict configuration similar to those in scenarios 6 and 7. The aircraft's speeds and headings are the same as



(a)



(b)

Figure 3.53. Conflict scenario 8. (a) Initial configuration (b) Predicted distances between the aircraft as they fly along their initially intended routes

in scenarios 6 and 7. However, in this scenario the ADS-B range of coverage is assumed to be 100 nm.

As it has been done for scenarios 6 and 7, the application of the mechanism in scenario 8 has been repeatedly simulated 50 times assuming that A_1's flight crew accepts its proposed resolution trajectory first in all the simulation runs. In this scenario, A_1 plans a resolution trajectory that allows A_3 to maintain its preferred route and simultaneously aims to induce A_2 to plan a resolution trajectory that resolves not only its conflict with A_1, but also its conflict with A_3. However, in 48 out of the 50 simulation runs A_2's resolution trajectory does not resolve its conflict with A_3 and, consequently, A_3 has to plan a resolution trajectory as soon as it detects the unresolved conflict, once A_2 and A_3 enter each other's ADS-B coverage area. In the other two simulation runs, the three conflicts are resolved without A_3 having to manoeuvre. Table 3.21 shows the means of the samples obtained in this case.

$\bar{x}$ Cost	A_1	2.49
	A_2	2.27
	A_3	2.69
	Total	7.46
$\bar{x}$ CPA distance (nm)	A_1-A_2	5.45
	A_1-A_3	8.91
	A_2-A_3	5.62

Table 3.21. Sample means of the resolution costs and the distances at CPA obtained from simulating 50 times the resolution of the conflicts in scenario 8 assuming that A_1 acts as the leader and both A_2 and A_3 accept their respective proposed resolution actions before they detect each other. In each simulation run, A_1's flight crew response latency is selected at random from the interval [40 s, 80 s]

Comparing Table 3.21 with Tables 3.20 and 3.19, it can be seen that the total resolution cost is shared more equitably among the three aircraft in scenario 8 than in scenarios 7 and 6. This is explained by the fact that in scenario 8 aircraft the three aircraft have to manoeuvre in almost all the simulation runs. It can also be observed that the sample mean of the total resolution cost is lower in scenario 8 than in scenarios 7 and 6. This can be justified by the fact that the ADS-B coverage range is longer in scenario 8.

An example of the application of the co-operation mechanism in conflict scenario 8 is shown below. In this example, both A_2 and A_3 have to manoeuvre to solve the three conflicts in the scenario. Assuming that A_1's flight crew response latency is 65 seconds and that both A_2's and A_3's flight crews accept their responses to A_1's resolution trajectory before they detect each other, the application of the mechanism results in the resolution trajectories depicted in Figure 3.54(a). Table 3.22 shows the defining parameters and the costs of the trajectories in Figure 3.54(a). As it can be seen in Figure 3.54(b), these trajectories do not resolve the conflict between A_2 and A_3. When A_2 and A_3 enter each other's ADS-B coverage area, A_3 receives A_2's resolution trajectory and detects the conflict between its intended route and that resolution trajectory. Hence, A_3 plans a resolution trajectory that is conflict-free with both A_1's and A_2's resolution trajectories. It is assumed that A_3's flight crew accepts this resolution trajectory 60 seconds after it detects its conflict with A_2's resolution trajectory. Figure 3.55(a) displays the final resolution strategy, which now includes A_3's resolution trajectory. Figure 3.55(b) shows the predicted distances between the aircraft as they fly their planned resolution trajectories. Table 3.23 shows the defining parameters and the costs of the trajectories in Figure 3.55(a).

Considering that the conflicting situation in scenario 8 is very similar to that in scenario 7, it is notable that a difference of 10 nm in the ADS-B range of coverage causes such radical differences between the performance of the co-operation mechanism in the two scenarios. These differences can be partially explained by the fact that longer ADS-B ranges of coverage result in longer times between the detection of a conflict and the closest point of approach, which in turn widens the range of resolution actions potentially capable of resolving the conflicts. In principle, the earlier the conflicts are detected, the wider the choice of resolution trajectories available for the conflicting aircraft to resolve them. In scenario 8, the fact that A_2 has a wide range of potential resolution trajectories from which to select its response may result in the emulation of A_2's planning process performed by A_1 not reflecting the actual process on board A_2. When this occurs, the resolution trajectory planned by A_2 has not been anticipated by A_1 and, therefore, it may not be conflict-free with A_3's initially intended route.

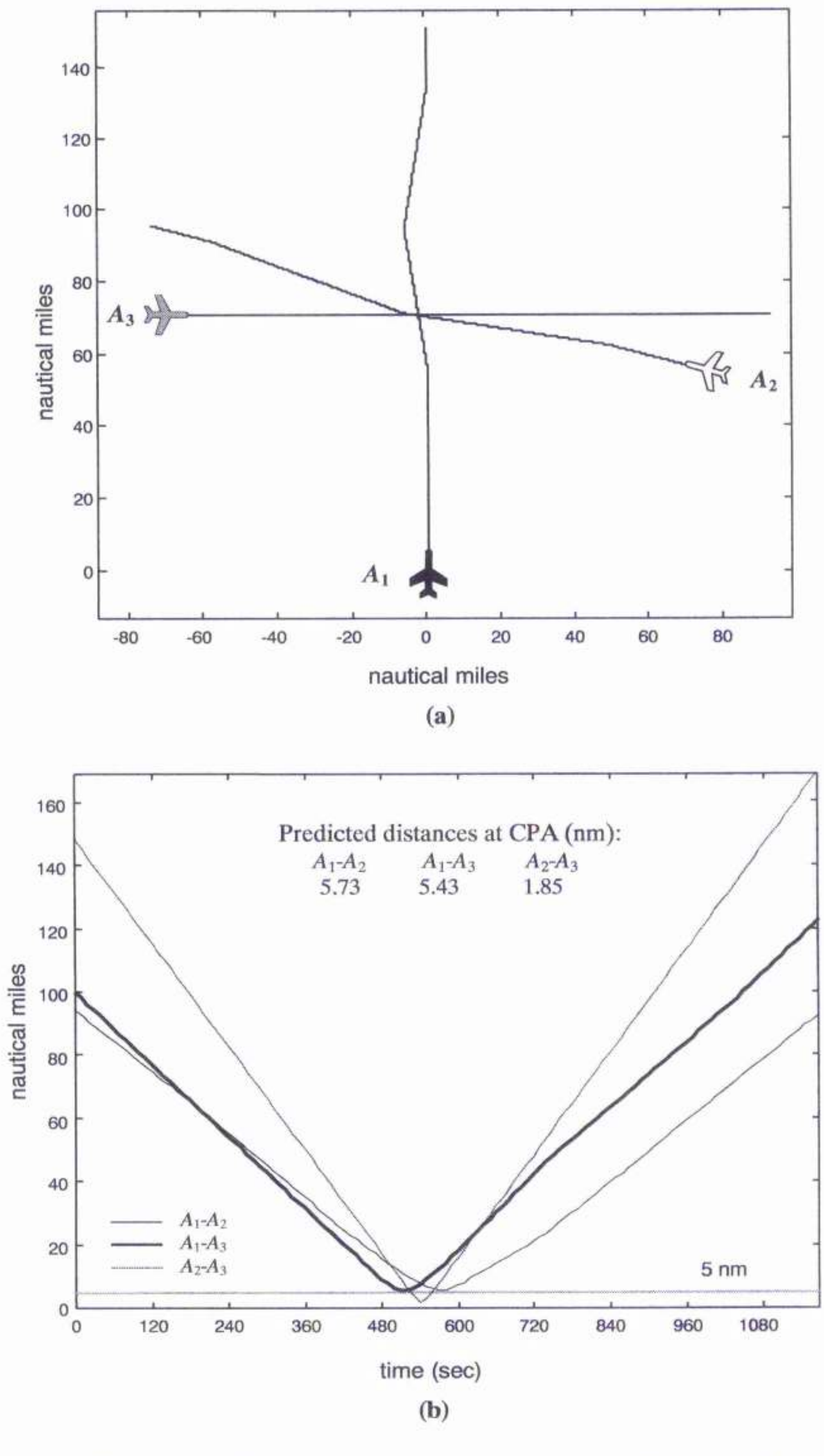
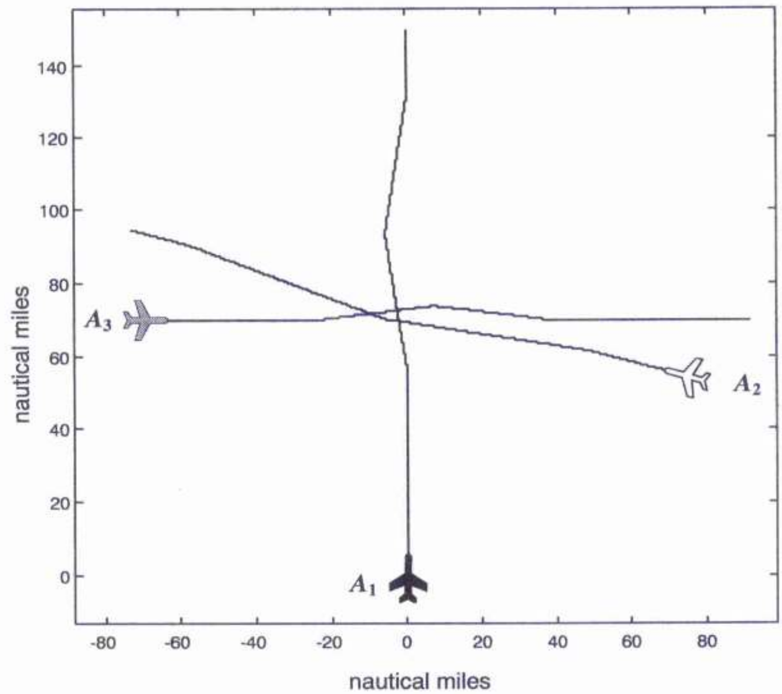
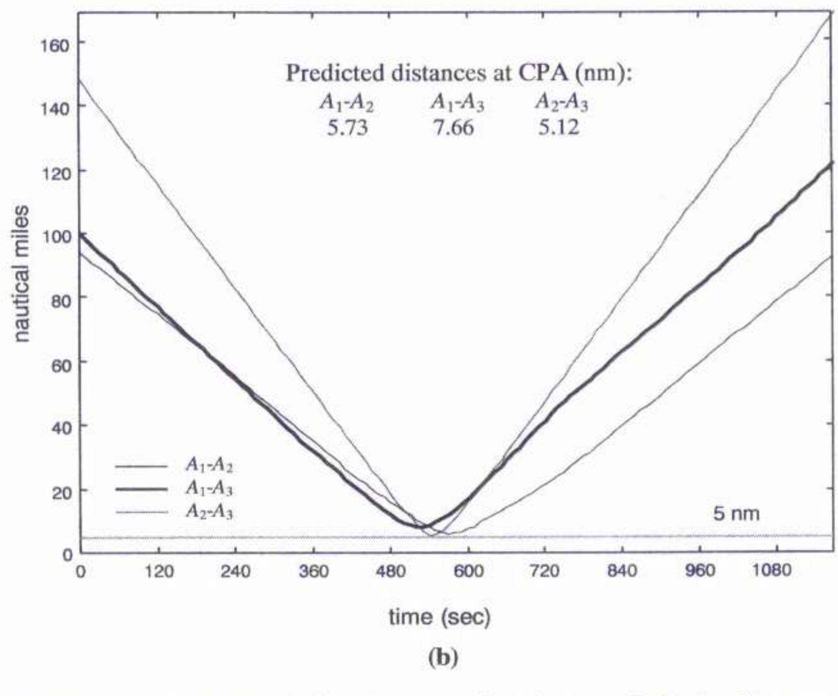


Figure 3.54. Simulation of the resolution of the conflicts in scenario 8. A_1 plans a resolution action that aims at inducing A_2 and A_3 to indirectly solve their conflict. A_1 acts first and its flight crew accepts the proposed resolution trajectory 65 s after it detects the conflicts. A_2's flight crew accepts the proposed resolution trajectory 55 s after receiving A_1's new intentions. A_3's flight crew accepts the proposed resolution trajectory 45 s after receiving A_1's new intentions. The resulting resolution strategy does not solve the conflict between A_2 and A_3. (a) Resolution trajectories accepted before A_2 and A_3 detect each other (b) Predicted distances between the aircraft if they flew those resolution trajectories



(a)



(b)

Figure 3.55. Simulation of the resolution of the conflicts in scenario 8. A_1 plans a resolution action that aims at inducing A_2 and A_3 to indirectly solve their conflict. A_1 acts first and its flight crew accepts the proposed resolution trajectory 65 s after it detects the conflicts. A2's flight crew accepts the proposed resolution trajectory 55 s after receiving A_1's new intentions. A_3's flight crew accepts the proposed resolution trajectory 45 s after receiving A_1's new intentions. Since the resulting resolution strategy does not solve the conflict between A_2 and A_3, A_3 produces a new resolution trajectory once it has detected A_2. A_3's flight crew accepts that trajectory 60 s after A_2 and A_3 have detected each other (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

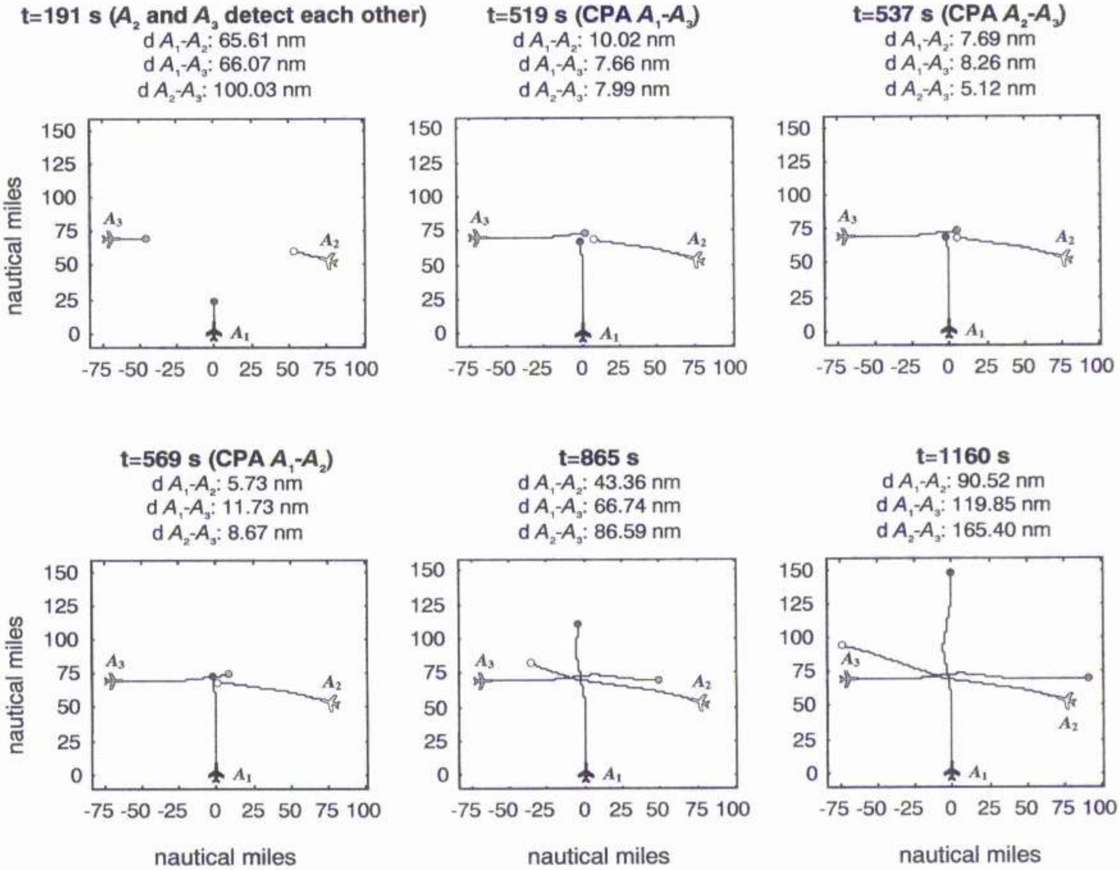


Figure 3.56. Simulation of the resolution of the conflicts in scenario 8: sequence of predicted future positions of the aircraft along their resolution trajectories in Figure 3.49(a)

	γ	t_s	t_e	Cost
A_1	-9°	432 s	1035 s	3.20
A_2	-6°	217 s	1045 s	2.10
A_3	Initially intended route			0.00
Total conflict resolution cost				5.30

Table 3.22. Simulation of the resolution of the conflicts in scenario 8: values of the defining parameters and costs of the resolution trajectories in Figure 3.54(a), which are planned before A_2 and A_3 detect each other

	γ	t_s	t_e	**Cost**
A_1	-9°	432 s	1035 s	3.20
A_2	-6°	217 s	1045 s	2.10
A_3	-7°	345 s	771 s	2.19
Total conflict resolution cost				7.49

Table 3.23. Simulation of the resolution of the conflicts in scenario 8: values of the defining parameters and costs of the resolution trajectories in Figure 3.55(a). A_3's resolution trajectory is planned after A_2 and A_3 detect each other

3.5 Conclusions and future work

The co-operation mechanism presented in this chapter exemplifies the potential of multi-agent systems concepts and techniques to support co-operative ADS-B-based airborne separation assurance in AAO. Autonomous Aircraft have been modelled as knowledge-based agents and conflicting aircraft-agents have been considered as constituting a multi-agent system, in the context of which they interact with one another to resolve their conflicts co-operatively. To achieve co-operation, conflicting aircraft-agents rely on a co-operation mechanism, which encompasses the algorithms, protocols and operational procedures that guide their actions regarding conflict resolution. Since the aircraft-agents can only interact with one another through the broadcast and reception of ADS-B messages, they are not capable of addressing one another to exchange information on a one-to-one basis and agree on a common resolution strategy. Consequently, the behaviouristic approach to co-operation in multi-agent systems has been adopted and a knowledge-based co-operation mechanism has been proposed. The proposed mechanism is based on the Recursive Modelling Method (RMM) and relies on the internal models that aircraft-agents build of one another. Each aircraft-agent is willing to contribute towards the resolution of the conflicts in which it is involved and considers that the other conflicting aircraft are also willing to do so.

The main element of the proposed co-operation mechanism is a trajectory-planning algorithm installed on board the aircraft-agents. The algorithm plans a resolution trajectory that is conflict-free with the intended resolution trajectories already broadcast by other conflicting aircraft, the conflicting leaders, and that is simultaneously expected to facilitate the resolution actions of the conflicting aircraft that remain to act, the conflicting followers. The resolution trajectories available to the conflicting aircraft are classified into a set of categories known to all the aircraft-agents. Each of these categories is represented by a conflict resolution pattern. The conflict resolution patterns are used by the trajectory-planning algorithm to anticipate the possible responses of the conflicting followers. The algorithm incorporates an iterative improvement process that allows for the planned resolution trajectory to be cost-efficient according to the aircraft-agent's criteria.

The trajectory-planning algorithm has been designed to be integrated into the ASAS on board equipment. The resolution trajectory produced by the algorithm is presented to the flight crew, which, in nominal operation, accepts it after a certain time lag. The algorithm estimates the flight crew response latency when planning the resolution trajectory. When the flight crew accepts the proposed trajectory, the new aircraft intent is broadcast through ADS-B and the aircraft becomes a conflicting leader of the conflicting aircraft that remain to act. This leader-follower structure serves as a means to co-ordinate the resolution actions of the conflicting aircraft.

The performance of the co-operation mechanism in two-dimensional conflict scenarios involving up to three aircraft has been investigated. In this types of scenarios, the available resolution trajectories have been restricted to lateral shift manoeuvres and have been grouped into three categories, each of them represented by a conflict resolution pattern. The flight crew response latency has been modelled with a uniform probability density function. The performance of the co-operation mechanism has been analysed statistically considering the probabilistic nature of the iterative improvement process and the randomness of the response latencies of the flight crews. In the scenarios studied, the mechanism has been shown to enable the conflicting aircraft to co-ordinate their actions so that they safely resolve their conflicts and share the resolution costs. From a multi-agent systems perspective, the mechanism provides the means for behaviouristic co-operation in conflict resolution, as the aircraft-agents can be said to act in a co-operative manner when resolving their conflicts.

Chapter 4

Reflective co-operation in Autonomous Aircraft Operations

4.1 Introduction

This chapter aims to demonstrate the potential of applying concepts and techniques used in the modelling and design of multi-agent systems, to develop co-operative conflict resolution methodologies for Autonomous Aircraft Operations in Operational Environment B. In this operational environment, which was introduced in chapter 2, Autonomous Aircraft are assumed to be able, not only to broadcast and receive ADS-B messages, but also to address one another and exchange additional information through a point-to-point data-link. To achieve its objective, this chapter proposes a co-operation mechanism that illustrates how the reflective approach to co-operation in multi-agent systems provides the means for the development of algorithms, operational procedures and communication protocols that support co-operative conflict resolution among Autonomous Aircraft in the operational environment considered.

The co-operation mechanism presented in this chapter is based on the reflective co-operation mechanism presented in [66]. Conflicting aircraft-agents, which are viewed as intentional systems, are deemed to co-operate when they are committed to resolve their conflicts through the implementation of a joint resolution plan. They exchange information with one another to form a team and agree on a joint plan that ensures the safe co-ordination of their resolution actions while allowing them to share the resolution costs equitably. The algorithms, procedures and protocols that constitute the mechanism, are designed to guide the flight crews' decision-making during the conflict resolution process. This guidance would be achieved through the ASAS equipment and the CDTI.

In the remainder of this chapter, Operational Environment B is described and, subsequently, the proposed co-operation mechanism is explained in detail and illustrated with examples.

4.2 Operational Environment B: Autonomous Aircraft Operations with point-to-point data-link communications

The only difference between Operational Environments B and A is that in the former the Autonomous Aircraft have the additional capability of communicating with each another on a one-to-one basis through a point-to-point data link. This point-to-point data-link facility allows the aircraft to address one another to reliably transmit and receive information not included in the ADS-B messages. While flying within AAO airspace, Autonomous Aircraft use this data-link to exchange information regarding the resolution of conflicts. Currently, no mature data-link technology is available to support such air-to-air point-to-point communications. Consequently, assumptions must be made concerning the future operational capabilities of such point-to-point data-link technology. For the purpose of this thesis, it is assumed that the data-link satisfies the communication requirements of the proposed co-operation mechanism as long as the distance between the aircraft is not greater than the ADS-B range of coverage. The proposed mechanism focuses on the information that the aircraft-agents must exchange to achieve co-operation and does not consider the possible limitations imposed by the data-link technology in place.

The description of Operational Environment A presented in section 3.2 of the previous chapter is applicable to Operational Environment B if expanded to include the fact that Autonomous Aircraft are equipped to communicate with one another through a high-performance point-to-point data-link. A list of the main features of Operational Environment B is shown in Table 4.1. Comparing Table 4.1 with Table 3.1 in chapter 3, which schematically describes Operational Environment A, it can be seen that they are virtually identical except for the shaded text in Table 4.1. This text is an addition to Table 3.1 and refers to the point-to-point data-link in Operational Environment B.

4.3 Co-operative conflict resolution in Operational Environment B

The reflective approach to co-operation in multi-agent systems has been adopted to model airborne separation assurance in Operational Environment B with the objective of developing a reflective co-operation mechanism that enables Autonomous Aircraft to resolve their conflicts through jointly agreed conflict resolution plans. According to the reflective approach, co-operation is deemed to occur when the agents are in a mental state that compels them to engage in joint activity aiming at the achievement of a common goal [66]. The proposed co-operation mechanism provides the means for the aircraft-agents to attain that mental state. This section first discusses how conflict resolution in AAO has been modelled according to the reflective approach to co-operation in multi-agent systems and then describes the co-operation mechanism proposed to implement that model of conflict resolution. The description of the co-operation mechanism focuses on how it would operate in a generic multi-aircraft conflict scenario. Subsequent sections will illustrate how the application of the proposed mechanism results in joint conflict resolution plans that ensure the co-ordination of the resolution actions of the conflicting aircraft as well as the equitable distribution of the resolution costs amongst them.

Airspace characteristics	
Airspace structure	• Airspace allocated to Autonomous Aircraft Operations. • Generally high altitude (above FL335). • No ATC coverage (possibly oceanic and over remote areas). • No fixed route structure: users fly their preferred routes between entry and exit points to Autonomous Aircraft operations airspace.
Traffic structure	• Generally en-route cruising traffic. • Possibly end-of-climb and top-of-descent traffic. • Likely crossing points between aircraft routes.
Separation assurance criteria	• As described in chapter 2, section 2.4.1.
Air Traffic Management support	• No ATC support: Separation assurance responsibilities fully delegated to the flight crew. • ATFM service: air traffic density has to be maintained below a certain level so that conflicts are manageable by ASAS.
Aircraft Equipment	
Communications	• Air-ground and air-air voice communications. • Air-ground data-link: exchange of information with the ground ATM centres. • Air-air point-to-point data-link capability: exchange of information with neighbouring aircraft on a one-to-one basis
Navigation	• GNSS equipment. Fully airborne navigation (no ground-based navigation aids).
Surveillance	• ADS-B equipment.
Traffic Situational Awareness and Separation Assurance	• Appropriate ASAS equipment: conflict detection and resolution support tools. • Pilot-ASAS interface, including CDTI and ASAS control panel.
Others	• 4D-FMS • ACAS mandatory.

Table 4.1: Schematic description of Operational Environment B. Shaded text indicates additions to Table 3.1

4.3.1 The reflective approach to co-operation applied to conflict resolution in Operational Environment B

As explained in chapter 2, adopting the reflective approach to co-operation implies taking the intentional stance towards agency. Accordingly, agents are thought of as intentional systems, which are entities who are ascribed beliefs, desires, intentions and other human mental attributes to explain their behaviour [60]. Regarding the legitimacy and usefulness of attributing mental qualities to artificial agents, McCarthy, among others, has argued that there are occasions when the intentional stance is adequate:

> "To ascribe *beliefs, free will, intentions, consciousness, abilities,* or *wants* to a machine is legitimate when such an ascription expresses the same information about the machine that it expresses about a person. It is useful when the ascription helps us understand the structure of the machine, its past or future behaviour, or how to repair or improve it. It is perhaps never logically required even for humans, but expressing reasonably briefly what is actually known about the state of the machine in a particular situation may require mental qualities or qualities isomorphic to them. ... Ascription of mental qualities is most straightforward for machines of known structure such as thermostats and computer operating systems, but is most useful when applied to entities whose structure is incompletely known." [94]

Thus, adopting the intentional stance towards agency is not just the result of the human tendency to anthropomorphism, but a potentially powerful approach to the analysis and designs of artificial agents. In fact, Singh, Rao and Georgeff state in [61] that the intentional stance is, in many cases, the most useful approach to agency. They argue that if an agent is given high-level cognitive specifications involving human mental attributes such as beliefs, desires and intentions, we will be able to "define the current state of [the] agent, what the agent might do, and how the agent might behave in different situations without regard to how the agent is implemented" [61].

Considering the foregoing, the intentional stance has been adopted when modelling Autonomous Aircraft as agents with the objective of defining a set of high-level cognitive specifications that result in the aircraft-agents resolving conflicts co-operatively. These high-level cognitive specifications are expressed in the form of the mental states needed for the aircraft-agents to engage in joint conflict resolution actions. Thus, the aircraft-agents are thought of as intentional systems and are ascribed human-like mental states that conceptually represent them and specify their behaviour. One of the advantages of adopting a mental state perspective is that it results in a conceptual model of the aircraft-agents that does not depend on their individual internal architectures or agent programs.

4.3.1.1 Adopting the intentional stance in modelling Autonomous Aircraft as agents: basic mental states

Having decided to take the intentional stance to describe aircraft-agents, the next stage is to provide a precise characterisation of their possible mental states. The aircraft-agents' mental states will be shown to guide their behaviour so that they resolve their conflicts through joint resolution plans.

As stated in chapter 2, any piece of information that can be thought of as known by the aircraft-agent is considered part of its knowledge base, regardless of where that piece of information is stored or how it is accessed. According to the intentional stance, the knowledge base is referred to as the aircraft-agent's set of beliefs [60]. A belief is a representation of a specific piece of the aircraft-agent's knowledge. The aircraft-agents are considered to be in a mental state of belief in relation to each of their beliefs so that they hold their beliefs as true. For example, if an aircraft-agent believes that its flying altitude is 30,000 ft, then it is persuaded that 30,000 ft is its true altitude.

The aircraft-agents are capable of communicating with one another to exchange information about their beliefs. For example, the broadcast of ADS-B messages enable aircraft-agents to share their beliefs about their position, speed and intended trajectory with surrounding aircraft-agents. Aircraft-agents are also able to use their computational capabilities to combine the information they receive through communication with their current beliefs with the objective of producing new pieces of information, which might in turn become part of the aircraft-agents' set of beliefs.

In addition to the mental state of belief, the mental states of *commitments* and *conventions* are also considered to describe the behaviour of the aircraft-agents [66]. A commitment is a pledge to pursue a certain objective. Aircraft-agents can make commitments about both beliefs and actions. For example, an aircraft-agent can make a commitment to fly to a certain waypoint WP_C and a separate commitment to fly a certain trajectory T_C to get to WP_C. Thus, the aircraft-agent has the goal of flying to WP_C and is committed to the achievement of that goal and, besides, it is also committed to pursue the achievement of its goal by flying a particular trajectory T_C. The former commitment refers to the aircraft-agent's intention of bringing about a state of affairs where it believes that its position is WP_C, regardless of the actions it might need to undertake to do so. The latter commitment refers to the aircraft-agent's pledge to perform a specific sequence of actions that is expected to achieve the desired state of affairs.

In principle, the aircraft-agents must endeavour to honour their commitments. However, commitments might be contradictory or incompatible among themselves, or might become unattainable due to a change in the external circumstances. For example, an aircraft-agent might be committed to arrive at waypoint WP_C before a specified time t_C and simultaneously committed to fly a trajectory that would cause the aircraft-agent to arrive at WP_C after t_C. To avoid such situations, the aircraft-agents must be provided with policies for governing the reconsideration of their commitments. These policies are incorporated into the mental states of conventions. A convention describes situations under which the aircraft-agent should re-assess a specific commitment. Should one of such situations arise, the convention indicates the course of action to be undertaken by the aircraft-agent. By means of an illustration, figure 4.1 shows an example of a possible convention for the commitment to fly a specific intended trajectory T_C.

4.3.1.2 Reflective approach to co-operative conflict resolution: the mental state of *Joint Responsibility*

Consider a generic conflict scenario involving *n* Autonomous Aircraft in which each is conflicting with at least one of the other *n*-1 aircraft. According to the reflective

```
CONVENTION FOR THE COMMITMENT TO FLY INTENDED TRAJECTORY Tc:

REASONS FOR RE-ASSESSING COMMITMENT:

•  Tc ALREADY FLOWN BY THE AIRCRAFT (COMMITMENT SATISFIED)
•  THE AIRCRAFT IS UNABLE TO FLY Tc (COMMITMENT UNATTAINABLE)
•  Tc CONFLICTS WITH A PROXIMATE AIRCRAFT'S INTENDED TRAJECTORY

ACTIONS:

RULE 1:      IF     Tc ALREADY FLOWN BY THE AIRCRAFT OR
                    THE AIRCRAFT IS UNABLE TO FLY Tc
             THEN   DROP COMMITMENT TO FLY Tc AND
                    ESTABLISH A NEW INTENDED TRAJECTORY

RULE 2:      IF     Tc CONFLICTS WITH A PROXIMATE AIRCRAFT'S
                    INTENDED TRAJECTORY
             THEN   DROP COMMITMENT TO FLY Tc AND
                    INITIATE CONFLICT RESOLUTION PROCCESS
```

Figure 4.1. Sample convention for the commitment to fly a specific intended trajectory

approach, co-operation is deemed to occur when the n aircraft-agents are in a specific mental state that compels them to form a team and engage in a joint conflict resolution action. With the objective of characterising such a mental state, the model of *Joint Responsibility*, which was proposed by Jennings in [66], has been adopted here. The model of Joint Responsibility provides a formal foundation for joint actions in multi-agent system from a mental state perspective. Following on from the model, if the n aircraft-agents are to pursue a joint conflict resolution plan P towards the achievement of the goal G, which is the resolution of all the conflicts in which they are involved, then the n aircraft-agents must adopt the mental state of Joint Responsibility. Before formulating a definition of the mental state of Joint Responsibility in this context, it is necessary to explain the concept of *mutual belief*, as it is fundamental in the model of Joint Responsibility and will be shown to be at the core of the definition.

A group of agents are said to mutually believe the preposition p if and only if every agent in the group believes p, every agent in the group believes that every agent in the group believes p, every agent in the group believes that every agent in the group believes that every agent in the group believes p, and so on *ad infinitum* ([66], [95]). A major problem of this definition of mutual belief is that it implies infinitely nested beliefs and, consequently, it is not achievable in practical terms, as it would require the agents to have an infinite amount of memory at their disposal. Thus, a more functional definition was proposed by Tuomela in [95] to describe mutual belief in practical settings. According to this definition, a group of agents can be said to mutually believe a proposition p when they have nested beliefs about that preposition up to a certain finite level so that they can be successful in their environment. For example, to ensure the orderly flow of traffic at a junction controlled by traffic lights, all drivers at the junction must believe that green means "go" and red means "stop". Further, each driver must believe that the other drivers at the junction believe that green means "go" and red means "stop", so that they can confidently act according to their beliefs about the colour coding of the traffic lights. The drivers can be said to mutually believe that green means "go" and red means "stop" despite the fact that they only have two levels of nesting of beliefs (everyone knows p and everyone knows that everyone knows p), as deeper levels of nesting are not required for the traffic at the junction to be satisfactorily controlled.

Jennings showed in [66] that a definition of mutual belief involving two levels of nesting of beliefs suffices to characterise the mental state of Joint Responsibility in an industrial multi-agent system so that the agents in the system can satisfactorily engage in joint activity. Such a definition of mutual belief has been adopted in the context of multi-agent systems comprised of conflicting aircraft-agents in AAO. Accordingly, a group of conflicting aircraft-agents are said to mutually believe that they have the goal G of resolving the conflicts in which they are involved when: (i) every aircraft-agent in the group has G as a goal (every aircraft-agent believes that it wants the conflicts to be eventually resolved), and (ii) every aircraft-agent in the group believes that every aircraft-agent in the group has G as a goal.

Considering the latter definition of mutual belief, the mental state of Joint Responsibility can now be defined for the generic conflict scenario introduced above. The n conflicting aircraft-agents are said to be in the mental state of Joint Responsibility if and only if the following conditions are met:

- The aircraft-agents mutually believe that they all have the goal G of resolving the conflicts in which they are involved.

- The aircraft-agents mutually believe that they all wish to collaborate with one another and act together as a team to realise G through the implementation of a joint resolution plan. Thus, G is a common goal of the aircraft-agents.
- The aircraft-agents agree on a joint resolution plan P to attain their common goal G and mutually believe that they are all committed to the execution of their respective individual action specified in P.
- The aircraft-agents have at their disposal a *social convention* that specifies the conditions under which they would reconsider their commitment to their action within P. This convention also describes how to act, both locally and towards their fellow team members, when they alter that commitment. The social convention provides the means for mutual support among the aircraft-agents during the implementation of P.

A co-operation mechanism provides the means for the n aircraft-agents to achieve the mental state of Joint Responsibility. The mechanism includes the communication protocols, operational procedures and algorithms needed for the aircraft-agents to be able to form a team and commit themselves to resolving their conflicts through a joint conflict resolution plan. It also includes the social convention relating to the aircraft-agents' commitment to their action within the joint conflict resolution plan.

To achieve the mutual beliefs about the goal G and the plan P required by the mental state of Joint Responsibility, the aircraft-agents rely on one-to-one communications through data-link. They need to exchange information regarding the conflicts to be resolved and the plan to resolve them so that they can understand their actions within the plan and commit to them. However, achieving mutual belief through communication may prove difficult in situations where message delivery is not guaranteed. As an illustration of this, consider the following example. An aircraft-agent A wants to form a team with another aircraft-agent B to co-operatively resolve the conflict in which they are involved. Suppose the aircraft-agents are not certain that their messages are always successfully delivered to their intended recipients. To form a team, A and B must mutually believe that both of them are willing to resolve their conflict through a joint plan. A sends B a message proposing the joint resolution of the conflict with the objective of attaining this mutual belief. Suppose that B successfully receives the message, decides to join the team and sends a message to A indicating the acceptance of its proposal. Since the successful delivery of messages is not guaranteed, B cannot know for certain whether or not A has received an acceptance from B. Consequently, B is not convinced that A believes that B is willing to join the team. Even if the acceptance message was successfully delivered to A, B would still require a message from A acknowledging receipt of B's acceptance message for mutual belief to be attained. Suppose A receives B's acceptance message and sends an acknowledgement message back to B. Again, A's acknowledgement message may not be successfully delivered to B, and thus mutual belief is still not guaranteed.

To overcome these difficulties, it is assumed that, in nominal operation, all point-to-point data-link messages are always successfully delivered to their intended recipients and that all the aircraft-agents mutually believe it.

4.3.2 Reflective co-operation mechanism for conflict resolution in Operational Environment B

This section describes a reflective co-operation mechanism that enables a group of conflicting aircraft-agents to attain the mental state of Joint Responsibility. This mechanism provides the means for the aircraft-agents to form a team and resolve the conflicts in which they are involved through the implementation of a joint resolution plan. The description focuses on how the mechanism would be applied in a generic conflict scenario. The generic conflict scenario considered involves n proximate aircraft-agents $A_1, A_2, \ldots, A_{n-1}, A_n$, each of which is in conflict with at least one of the other n-1 aircraft-agents. Some of the n aircraft-agents may be outside the ADS-B range of coverage of one another, but there are k aircraft-agents, with $k \le n$ and $k \ge 1$, that are capable of detecting all the conflicts in the scenario. Each of these k aircraft-agents is within the ADS-B range of coverage of all the other n-1 aircraft-agents. The n-k aircraft-agents that do not have complete knowledge of all the conflicts in the scenario may be involved in conflicts that they are not able to detect.

4.3.2.1 Team formation process

Whenever an aircraft-agent A_i, with $i \in \{1, 2, \ldots, n-1, n\}$, believes it is in conflict with at least one of the other n-1 aircraft-agents, it attempts to team up with the aircraft-agents within its ADS-B coverage range that it believes are also involved in at least one conflict. These aircraft-agents are referred to as A_i's *potential team mates*. The number of potential team mates of A_i is denoted by l_i, with $l_i \leq n-1$. A_i sends a *team membership proposal message* to each of its l_i potential team mates requesting them to team up with it.

It is assumed that once an aircraft has received a team membership proposal message, it does not initiate the formation of another team unless its potential team mates include all the aircraft-agents in the list of members of the team already proposed and at least one more aircraft-agent not included in the list. Since the aircraft-agents are aware of the fact that they may not have complete knowledge of the conflicting situation, they consider all the team membership proposals that they either receive or issue during a prescribed time span of length t_f. After this time span has expired, the aircraft-agents proceed to form the team with the highest number of members among those proposed. In principle, the longer the prescribed time span, the more likely it is that the team formed will encompass all of the n conflicting aircraft-agents. However, an excessively long time span could substantially delay the team formation process and, consequently, the establishment of a joint conflict resolution plan. This could result in unacceptably complex and costly resolution manoeuvres. Thus, the length t_f of the prescribed time span represents a trade-off between the advantages of forming a team as soon as possible to allow for cost-effective resolution manoeuvres and the risks of forming a team that does not encompass all of the n conflicting aircraft.

The team formation process must be understood and sanctioned by the flight crews. Thus, once the prescribed time span expires, the aircraft-agents make their respective flight crew aware of the conflicting situation and of their intention to join a team to co-operatively resolve the conflicts of all the aircraft in that team. They also let their flight crew know the identities of their potential team mates, indicating which one has issued the team proposal, and request their permission to join the team. In nominal operation, the flight crew grants permission to join the team after a random time lag. As soon as the flight crew of an aircraft-agent that has received a team membership proposal message accepts to join the team, an acceptance message is sent to the aircraft-agent that issued the team proposal. Once this aircraft-agent has received an acceptance message from each of its potential team mates and its flight crew has accepted to join the team, it sends a message back to each of them confirming that the team has been established. The aircraft-agent that issues a team proposal that eventually results in the formation of a team is known as the *team organiser*. Once the team has been formed, the team organiser is responsible for determining a joint conflict resolution plan and transmitting to its team mates their respective actions within the joint plan.

Suppose that A_i is the first aircraft-agent sending a team membership proposal message to its l_i potential team mates. If A_i is one of the k aircraft-agents that are within the ADS-B range of coverage of all of the other n-1, then $l_i = n-1$ and the team proposed by A_i comprises of the n conflicting aircraft. None of the recipients of the team membership proposal messages sent by A_i would attempt to form a team themselves, as the aircraft-agents that they regard as potential team mates are already included in the team proposed by A_i. After a time span of length t_f since the team membership proposal was issued, A_i and its potential team mates request permission from their respective flight crew to join the team proposed by A_i. Again, it is assumed that, in nominal operation, the aircraft's flight crews accept joining the team after a certain random time lag. Subsequently, each of A_i's potential team mates sends a message to A_i accepting to join its proposed team. Then A_i sends a message back to each of them confirming that the team has been established. Hence, A_i, which is the team organiser, has successfully brought together a team that comprises of all the n conflicting aircraft.

If A_i is not one of the k aircraft-agents that are within the ADS-B range of coverage of all of the other n-1, then $l_i < n-1$ and the team proposed by A_i does not comprise of all the n conflicting aircraft. Hence, some of A_i's potential team mates may attempt to form a team that includes aircraft-agents that are not in the team proposed by A_i. Additionally, some of the $(n\text{-}1)\text{-}l_i$ aircraft-agents that have not received a team membership proposal message from A_i may also initiate a team formation process. Eventually, one of the k aircraft-agents that are within ADS-B coverage of all the other n-1 aircraft agents, which will be denoted by A_j with $1 \leq j \leq n$, sends a team membership proposal message to each of those n-1 aircraft-agents. From then on, no more team membership proposals are issued, as the team proposed by A_j comprises of all of the n conflicting aircraft. Each of A_j's potential team mates eventually sends a message back to A_j accepting to join its proposed team. Subsequently, A_j, which is the team organiser in this case, sends a message back to each of them confirming that the team has been established. It is implicitly assumed that the chosen value of t_f allows for the n-1 potential team mates of A_j to receive a team membership proposal message from A_j before the end of the time span t_f since they first received or issued a team membership proposal.

When the n conflicting aircraft-agents are brought together into a team by a team organiser, they mutually believe that they have the goal of resolving all the conflicts in which they are involved. Although the team organiser is aware of all of these conflicts, some of its team mates might be outside of one another's ADS-B coverage and thereby unable to detect the conflicts among them. The n team members mutually believe that they all wish to collaborate with one another and act together as a team to accomplish their common goal through the implementation of a joint resolution plan, which is to be determined by the team organiser. Thus, the formation of a team implies an element of trust on the part of the n-1 aircraft-agents that accept the team organiser's proposal.

The team formation process must also incorporate the necessary operational procedures to cope with non-nominal situations such as those arising when an aircraft-agent does not accept to join the proposed team due to, for example, equipment failure or distress in the cockpit. The definition of these procedures is considered out of the scope of this thesis. Nevertheless, some suggestions about how such non-nominal situations could be tackled are given next. If the aircraft-agent not joining the team were not the team organiser, a team comprising of all of the n aircraft-agents except that one could be formed. In such situation, the team organiser would determine a joint plan that resolves all the conflicts without requiring that aircraft to modify its intended route. If the aircraft-agent not joining the team were the team organiser, then the team formation process would have to be discarded. Another aircraft-agent within the ADS-B range of coverage of all the others could initiate the process of forming a team comprising of all the n aircraft-agents except the one that did not join the original team, which would be assumed to maintain its initially intended route.

As an illustration of the team formation process explained above, consider an example of its application in conflict scenario 7. This conflict scenario was introduced in the previous chapter and is reproduced below in Figure 4.2. The ADS-B range of coverage is assumed to be 90 nm. Figure 4.2(a) depicts the positions of the three aircraft when A_1 and A_3 enter each other's ADS-B coverage area. A_1 and A_2 have been within each other's ADS-B coverage range for approximately 20 seconds and A_2 and A_3 will still be outside each other's ADS-B coverage for approximately 200 more seconds. Figure 4.2(b) shows that if the three aircraft maintain their currently intended route, each of them will conflict with the other two. When the aircraft positions are those shown in Figure 4.2(a), both A_2 and A_3 detect their respective conflict with A_1 but they are unable to detect the conflict with each other. A_1 detects the three conflicts in the scenario. As soon as A_1 and A_2 enter each other's ADS-B coverage area and detect the conflict between them, they attempt to team up with each other to resolve it in a co-operatively manner. Subsequently, once A_3 and A_1 enter each other's ADS-B coverage area, A_3 also attempts to team up with A_1 to resolve their conflict, while A_1, which is now aware of all the conflicts in the scenario, attempts to form a team comprising of the three aircraft.

The length of the time span during which an aircraft-agent has to consider all the team membership proposals that it either receives or issues before deciding which team to join is assumed to be t_f=30 seconds. It is also assumed that the flight crew response latency to a request for permission to join a team is a random time uniformly distributed between 40 seconds and 80 seconds. Thus, the PDF of the flight crew response latency in this case is the same as the one used to model the flight crew response latency in the previous chapter, which was depicted in Figure 3.6 and is reproduced in Figure 4.3. During the latency interval the flight crew becomes aware of the conflict situation as seen from its aircraft perspective and understands the team formation process that is taking place.

Suppose A_2 is the first aircraft-agent to issue a team proposal. In addition to A_2's proposal, both A_1 and A_2 have to consider any other team membership proposal that they either receive or issue during a time span of 30 seconds following the sending of A_2's team membership proposal message to A_1. After this time span, they propose their respective flight crews to join the team with most members among those proposed.

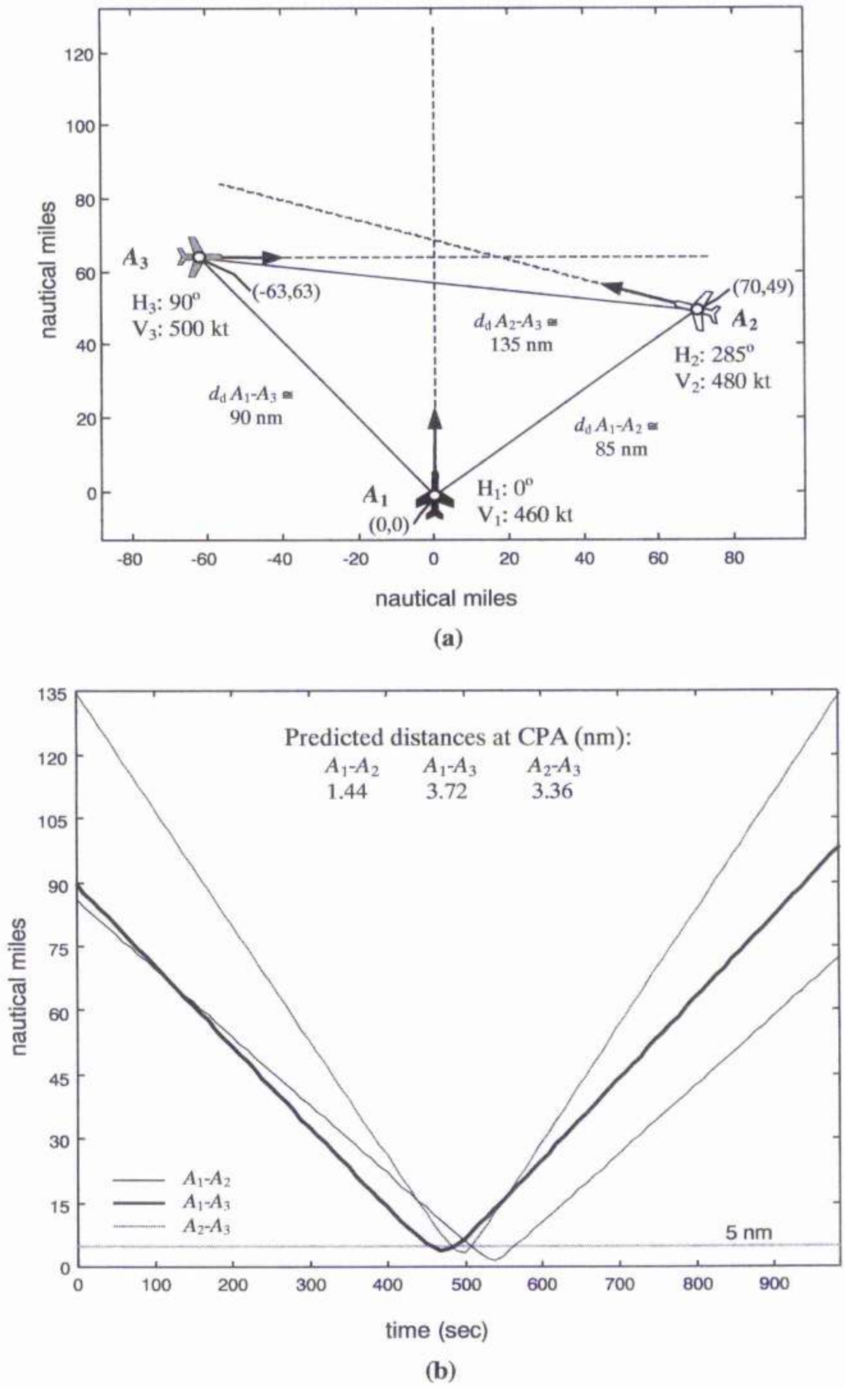


Figure 4.2. Conflict scenario 7. (a) Initial configuration (b) Predicted distances between the aircraft as they fly along their initially intended routes

Once A_1 and A_3 enter each other's ADS-B coverage area, which occurs before the prescribed 30 seconds time span has expired, A_1 decides to discard A_2's team proposal and start forming a larger team including both A_2 and A_3. Simultaneously, A_3 attempts

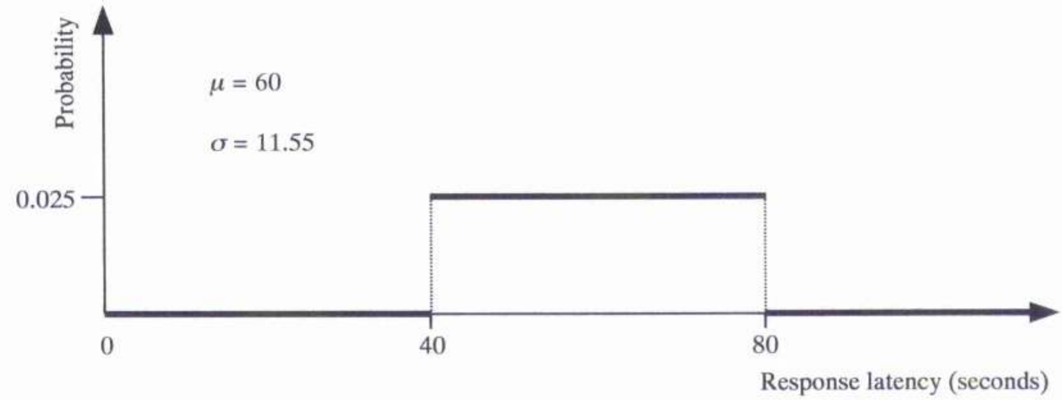


Figure 4.3: Probability density function of the flight crew response latency for team formation

to team up with A_1 to co-operatively resolve the conflict between them. Suppose A_3 receives a team membership proposal message from A_1 before issuing its own team proposal. Since A_3's team proposal does not include A_2, A_3 discards it and never sends a team membership proposal message to A_1. Before deciding which team to join, A_3 still has to consider any team membership proposal that it may receive or issue during a time span of 30 seconds following the reception of A_1's team membership proposal message.

When A_2 and A_3 receive A_1's team membership proposal message, they realise that they do not have complete knowledge of the conflicting situation and, unless larger teams are proposed, they will eventually decide to join the team proposed by A_1. In this example, the largest possible team is the one proposed by A_1 and, consequently, no more team proposals are issued after A_1's. After its respective 30 seconds time span has expired, each of the three aircraft-gents informs its respective flight crew of the conflicting situation and requests its consent to join the team proposed by A_1. While both A_1 and A_2 do so 30 seconds after A_2 initiates the team formation process, A_3 does so 30 seconds after A_1 sends its team membership proposal. Since A_1 and A_3 enter each other's ADS-B coverage area approximately 20 seconds after the start of the team formation process, A_3 requests permission from its flight crew to join the team approximately 20 seconds after A_1 and A_2. Suppose that A_1's, A_2's and A_3's flight crew response latencies are, respectively, 50 seconds, 70 seconds and 65 seconds. Thus, A_1's flight crew accepts joining the team approximately 80 seconds after the start of the team formation process, A_2 sends its acceptance message to A_1 approximately 100 seconds after the start of the team formation process and A_3 does so approximately 115 seconds after the start of the team formation process. This implies that the team is established approximately 115 seconds after A_2 starts the team formation process and approximately 95 seconds after the aircraft positions are those shown in Figure 4.2(a). A_1 is the team organiser and the team comprises of the three conflicting aircraft in the scenario. Figure 4.4 shows the time line of the example of team formation process described above.

Although it has been shown that the proposed team formation process enables the three conflicting aircraft in the example above to bring together a team involving the three of them, the length of the team formation process may hinder the satisfactory resolution of the conflicts. Considering that once the team has been formed a joint conflict resolution plan has still to be established with the consent of the flight crews, an excessively long team formation process could result in the impossibility to implement a satisfactory joint plan in time to resolve the conflicts. In principle, the shorter the look-ahead time for conflict detection, the more likely that the length of the team formation process prevents the aircraft from resolving their conflicts through a joint resolution plan. Consequently, the problem would be accentuated with shorter ADS-B ranges of coverage. As it has been shown in the example above, the chosen value of t_f and the response latencies of the flight crews determine the length of the team formation process. Hence, it is patent that the practical feasibility of the co-operation mechanism will depend on those two factors. This point will be discussed further later in this chapter when presenting the examples of the application of the co-operation mechanism.

4.3.2.2 Establishing a joint conflict resolution plan and monitoring its implementation

Once the team organiser has succeeded in bringing together a team comprising of all the n conflicting aircraft, it must elaborate a joint conflict resolution plan P and transmit to each of its team members its respective resolution action within that plan. To elaborate P, the team organiser has a planning algorithm at its disposal that assigns each team member a new intended trajectory. The output of the planning algorithm consists of a set of parameters defining the new intended trajectories of each of the n conflicting aircraft. These new intended trajectories are designed so that they anticipate the resolution of all the conflicts in which the team members are involved and the equitable

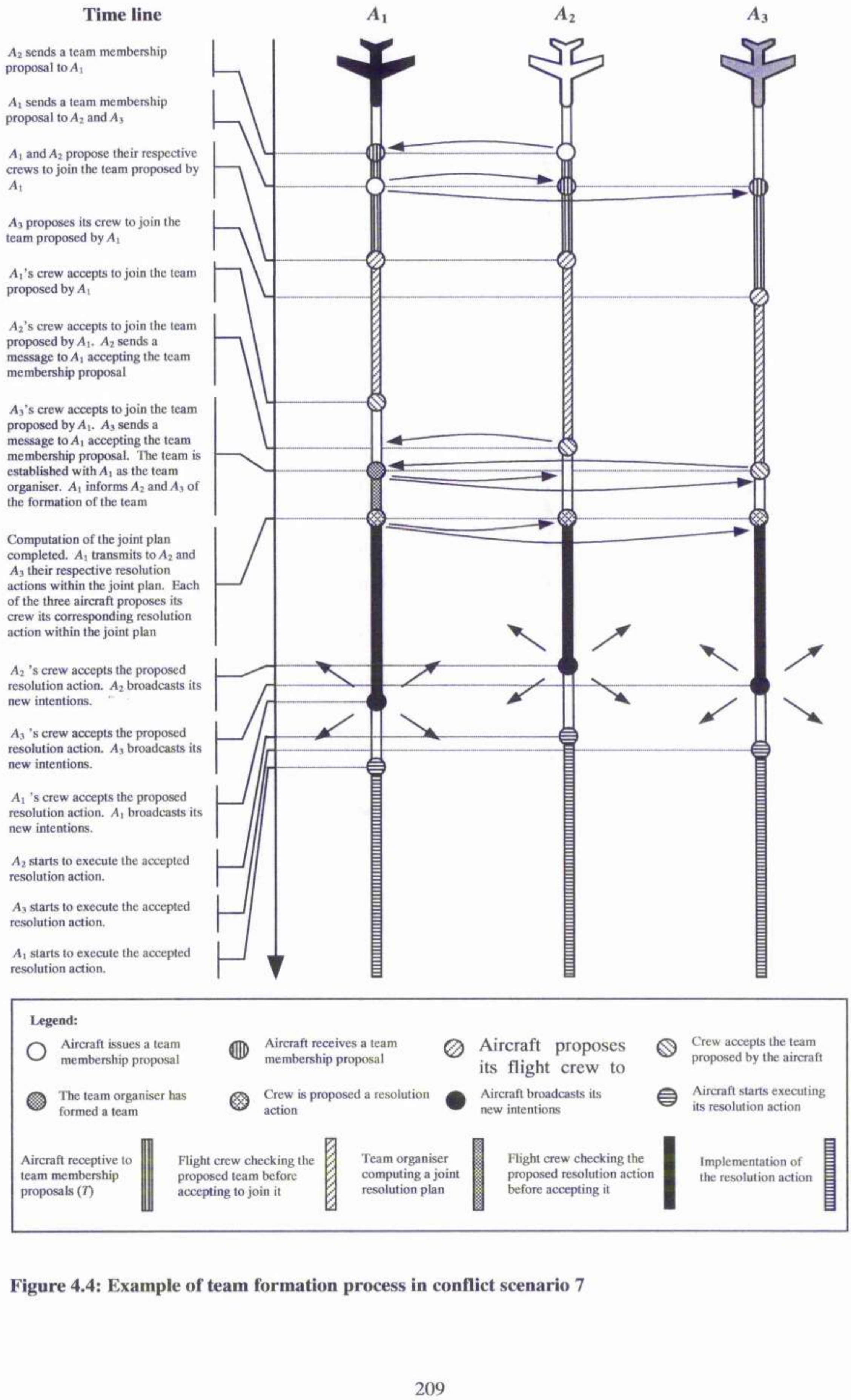


Figure 4.4: Example of team formation process in conflict scenario 7

distribution of the resolution costs amongst the team members.

The planning algorithm requires the organiser to have some knowledge about the performance characteristics of its team mates to ensure that they are able to fly their respective trajectories within the plan. Besides, the team organiser might also have to consider the cost-efficiency criteria of each of its team mates to achieve a fair distribution of the resolution costs among all the team members. The team organiser can store performance and cost-efficiency data about its team mates in its memory but it might be necessary for them to send it additional information through the point-to-point data-link once the team has been formed. This additional information, relating to performance limitations and cost-efficiency preferences, would contribute to ensure that the joint plan consists of flyable and efficient trajectories for all the team members. However, the transmission of this information might present some practical difficulties. First of all, the volume of information transmitted will be limited by the capacity of the data-link in place and the need to establish a joint plan in a timely manner. In addition, airlines might be reluctant to accede to the transmission of information regarding their cost-efficiency criteria to aircraft operated by other airlines for reasons of commercial competition. To make the use of such information by a team organiser acceptable to airlines, the transmissions could be encrypted and the team organiser could be made to delete from its memory the information transmitted once it has been used in the elaboration of the joint plan.

Suppose that, in nominal operation, the team organiser receives sufficient information from its team mates to elaborate a conflict-free joint plan P that assigns to each team member a flyable resolution trajectory that is not excessively costly according to its own efficiency criteria. Once it has determined P, the team organiser transmits to each if its team mates a set of parameters that define its respective resolution trajectory within P. It is assumed that the aircraft-agents are able to re-construct unambiguously the trajectory defined by the parameters received from the team organiser. The concept of a *trajectory language*, which was briefly discussed in section 3.2 of the previous chapter, might be applicable in this context. An advantage of encoding the resolution trajectories within P using a trajectory language known to all aircraft-agents is that different planning algorithms may be used. Whenever an aircraft-agent acting as a team organiser applies a planning algorithm to determine P, the trajectory language ensures that its team mates will understand their respective resolution trajectories regardless of how they have been produced.

As soon as a team member becomes aware of its resolution trajectory within P, it commits to its implementation. Since the messages sent by the team organiser are guaranteed to be successfully delivered to their intended recipients, once all the team members are aware of their respective resolution trajectory they mutually believe that each of them is committed to implementing it. Despite the fact that the resolution trajectories are in principle flyable and reasonably cost-efficient, they must be sanctioned by the flight crews before they are actually implemented by the aircraft-agents. Each team member presents its resolution trajectory to its flight crew, which, in nominal operation, accepts it after a random time lag. It has to be ensured that the resolution trajectory does not require the currently intended route to be modified before the flight crew has acceded to the implementation of the resolution trajectory.

It is assumed that, in nominal operation, all the team members fulfil their commitment to implementing its respective resolution trajectory within P and achieve their common goal G of resolving all the conflicts in which they are involved. The team members have a social convention at their disposal to monitor the implementation of the joint plan P and to ensure mutual support among them in non-nominal situations. Consider for example a non-nominal situation in which an aircraft-agent is unable to comply with the resolution trajectory that it has been assigned by the team organiser. The aircraft-agent would have to drop its commitment to the implementation of that trajectory, inform the team organiser and decide a different course of action. The social convention would guide the aircraft-agent's decision-making in such situation. Similarly, a non-nominal situation where an aircraft-agent's flight crew does not accede to the implementation of the resolution trajectory that it has been assigned by the team organiser would also require the application of the social convention. Again, the aircraft-agent would have to drop its commitment to flying its assigned resolution trajectory and determine a new course of action. Another example of a non-nominal situation in which the social convention is needed is when an aircraft-agent becomes aware, once it is within the ADS-B coverage area of all the other team members, that its resolution trajectory is not conflict-free with those of the other team members. This situation can be caused by an error in the elaboration of the joint plan, which in turn might be due to errors in the team organiser's knowledge about its team mates.

An example of a social convention for a team of aircraft-agents is shown in Figure 4.5. This social convention enables the team members to cope with the non-nominal situations mentioned above. The rules in this convention should have to be defined in detail for a practical implementation of the co-operation mechanism.

SOCIAL CONVENTION FOR A TEAM OF AIRCRAFT-AGENTS

REASONS FOR RE-ASSESSING COMMITMENT TO THE IMPLEMENTATION OF

MY RESOLUTION TRAJECTORY WITHIN *P* (*RT*):

- *RT* CANNOT BE IMPLEMENTED PROPERLY
- FLIGHT CREW DOES NOT ACCEDE TO THE IMPLEMENTATION OF *RT*
- *RT* IS NOT BEING IMPLEMENTED PROPERLY
- *RT* WILL NOT RESOLVE ALL MY CONFLICTS

ACTIONS:

RULE 1:	IF	*RT* CANNOT BE IMPLEMENTED PROPERLY OR FLIGHT CREW DOES NOT ACCEDE TO THE IMPLEMENTATION OF *RT* OR *RT* IS NOT BEING IMPLEMENTED PROPERLY OR *RT* WILL NOT RESOLVE ALL MY CONFLICTS
	THEN	DROP COMMITMENT TO *RT* AND INFORM TEAM ORGANISER
RULE 2:	IF	DROP COMMITMENT TO *RT*

Figure 4.5. Example of a possible social convention for a team of aircraft-agents

The performance of the reflective co-operation mechanism described above depends on the planning algorithm used by the team organiser. Unlike the behaviouristic co-operation mechanism presented in the previous chapter, which requires all the aircraft-agents to apply the same trajectory-planning algorithm, the proposed reflective co-operation mechanism does not specify a particular planning algorithm to be used by the team organiser. In principle, the team organiser might use any planning algorithm as long as it complies with certain minimum requirements. Among those minimum requirements, the following are proposed as the most important:

- The planning algorithm in place must enable the team organiser to elaborate a joint plan that consists of feasible conflict resolution trajectories.
- The resolution trajectories that constitute the joint plan must be expressible according to a standard trajectory language so that they are understandable to the team members.
- The joint plans produced by the algorithm are expected to distribute the conflict resolution costs among the team members.

The following section will present an example of a planning algorithm that complies with the above requirements and could be used by team organisers when applying the reflective co-operation mechanism proposed in this chapter.

4.4 Example of a planning algorithm for the reflective co-operation mechanism

This section presents a planning algorithm suitable for use within the framework laid out by the reflective co-operation mechanism described above. The proposed algorithm is designed to enable a team organiser to search for a combination of resolution manoeuvres that enables the team members to share equitably the costs of resolving the conflicts in which they are involved. The algorithm recasts the planning process in terms of a *multi-objective optimisation problem* ([96], [97], [98], [99]). This problem is subsequently converted to a single-objective *combinatorial optimisation problem*, which is tackled using a *metaheuristic technique* ([100], [101], [102], [103], [104]).

The remainder of this section is structured as follows. First, the planning algorithm is described for a generic conflicting configuration involving n aircraft-agents. Subsequently, the performance of the reflective co-operation mechanism with the proposed planning algorithm is analysed for two-dimensional conflict scenarios involving up to three aircraft flying along straight lines at constant speed. To do so, the planning algorithm has been specifically adapted to such type of conflict scenarios and implemented using the MATLAB® computing language. Considering the performance analysis of the behaviouristic co-operation mechanism conducted in the previous chapter, the performance of the two co-operation mechanisms in the two-dimensional conflict scenarios considered are compared.

4.4.1 Formulation of the planning process as a multi-objective optimisation problem

Suppose that the n aircraft-agents $A_1, A_2, \ldots, A_{n-1}, A_n$ have established a team to resolve the conflicts in which they are involved. Let the team member A_{i^*}, where $1 \le i^* \le n$, be the team organiser. A_{i^*} is responsible for determining a joint plan P consisting of a combination of n resolution trajectories, each of which is assigned to a team member. Each of the resolution trajectories in P is encoded in a message and transmitted to the team member to which that trajectory is assigned. The recipient of the message must be able to decode its assigned resolution trajectory and implement it accurately. The implementation of P must resolve all the conflicts in which the team members are involved. In addition, the joint plan P is expected to result in the fair distribution of the resolution costs among the team members.

Each of the resolution trajectories that constitute the joint plan P must be, first of all, feasible for the aircraft-agent to which it is assigned. Besides, the resolution trajectories should be designed so that they meet the aircraft-agents' cost-efficiency criteria and do not cause them to deviate excessively from their initially intended routes. Consequently, A_{i^*} has to search for a joint plan among all the possible combinations of feasible resolution trajectories for the team members taking into account their individual cost-efficiency criteria. To do so, A_{i^*} first of all defines a set of allowable resolution trajectories for each of the team members. A_{i^*} regards the resolution trajectories in each of these sets as feasible for the corresponding aircraft-agent.

Each of the allowable resolution trajectories of an aircraft-agent is mapped to a set of m real parameters, which is assumed to identify univocally the resolution trajectory. This mapping corresponds to expressing the allowable resolution trajectories using the common trajectory language known by all the team members. The set of the allowable resolution trajectories of an aircraft-agent A_i, with $i \in \{1, 2, \ldots, n\text{-}1, n\}$, is denoted as

RT_i^A. The mapping that assigns each resolution trajectory in RT_i^A to a set of m real parameters according to the trajectory language is denoted by $L: RT_i^A \rightarrow X_i$, where X_i is a subset of $\mathbf{R}^m$, with $\mathbf{R}$ denoting the set of real numbers. According to the mapping L, each resolution trajectory in RT_i^A is assigned a unique m-dimensional vector $\boldsymbol{x}_i=(x_i^1,\ x_i^2,\ldots,\ x_i^{m-1},\ x_i^m)$ that belongs to X_i. Considering the foregoing, an allowable joint plan P can be defined as an element $\boldsymbol{x}$ of a set X formed as the Cartesian product of the sets $X_1, X_2,\ldots, X_{n-1}, X_n$:

$$P \equiv \boldsymbol{x}=(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{n-1}, \boldsymbol{x}_n) \in X\,, X = X_1 \times X_2 \times \ldots \times X_{n-1} \times X_n \tag{4.1}$$

The elements of the set X can be expressed as vectors of dimension $m{\cdot}n$:

$$\boldsymbol{x}=(x_1^1,\ldots,\ x_1^m, x_2^1,\ldots,\ x_2^m,\ldots\ldots,\ x_{n-1}^1,\ldots,\ x_{n-1}^m, x_n^1,\ldots,\ x_n^m) \tag{4.2}$$

In addition to defining the allowable joint plans, A_{i^*} encodes its knowledge about the team members' cost-efficiency criteria in *cost functions*. The cost function for the team member A_i, with $i \in \{1, 2,\ldots, n\text{-}1, n\}$, is a real-valued function, C_i: $Xi \rightarrow \mathbf{R}$, that assigns each allowable resolution trajectory of the aircraft-agent A_i to a real number representing the cost that the resolution trajectory induces on A_i. It is assumed that the cost functions defined by the team organiser are of the form:

$$C_i(\boldsymbol{x}_i) = a_i^1 c_1 + a_i^2 c_2 + \ldots + a_i^{q-1} c_{q-1} + a_i^q c_q \tag{4.3}$$

where $c_1, c_2,\ldots,c_{q-1}, c_q$ are pre-defined standard cost functions, which may relate to factors such as fuel consumption or additional flight time, and $a_i^1,\ a_i^2,\ldots,\ a_i^{q-1},\ a_i^q$ are parameters that indicate the individual preferences of A_i regarding the standard cost functions.

Given the individual cost functions of the n team members, the team organiser A_{i^*} can define a series of n cost functions assigning each allowable joint plan the cost that it entails for each of the team members. The i^{th} function in the series, which is denoted as

$C_i^X : X \rightarrow \mathbf{R}$, is defined over the set X and assigns each allowable joint plan $x \in X$ the cost of A_i's resolution trajectory within that joint plan:

$$C_i^X(x) = C_i(x_i);\ i=1, 2, \ldots, n-1, n;\ x=(x_1, x_2, \ldots, x_{n-1}, x_n) \in X \tag{4.4}$$

These n functions are seen as the components of the following vector-valued function:

$$C^X: X \rightarrow \mathbf{R}^n;\ C^X(x) = (C_1^X(x), C_2^X(x), \ldots, C_{n-1}^X(x), C_n^X(x)) \tag{4.5}$$

Considering the foregoing, the process of searching for a joint plan that resolves all the conflicts in which the team members are involved while distributing the resolution costs equitably among them will be recast below as a multi-objective optimisation problem.

4.4.1.1 Multi-objective optimisation and Pareto optimality

When thinking of optimisation in general, one usually thinks about the problem of minimising or maximising a single well-defined objective, which is a quantitative measure of the performance of the system under study. This objective is usually a cost or utility function defined over a set of candidate solutions, which may have to comply with certain constraints. In these single-objective optimisation problems, the candidate solutions may be ordered unambiguously according to the cost or utility function. The aim of the optimisation in this context is to find the highest-ordered candidate solution.

Multi-objective optimisation is concerned with problems involving several cost or utility functions, which are possibly conflicting and expressed in different units. In this type of problems, establishing an ordering of the candidate solutions is not straightforward and the concept of Pareto optimality ([96], [97], [98], [99]) is usually applied. According to this concept, the optimisation process aims to find a set of trade-offs among the different objectives, instead of searching for a single optimum. These trade-offs are referred to as *Pareto-optimal solutions*, which are optimal in the sense that no other candidate solutions can improve upon an objective without causing a detrimental effect in at least one other objective.

Consider the problem of minimising simultaneously n real-valued cost functions defined over a set Y of candidate solutions. Suppose that the elements of the set Y are vectors of dimension $m{\cdot}n$ that satisfy certain given constraints. This problem can be expressed as follows:

$$\begin{aligned} &\textit{minimise} \;\; \mathbf{g}(\mathbf{s}) = (g_1(\mathbf{s}), g_2(\mathbf{s}), \ldots, g_{n-1}(\mathbf{s}), g_n(\mathbf{s})) \\ &\textit{subject to} \;\; \mathbf{s} \in Y \end{aligned} \tag{4.6}$$

where $\mathbf{s}$ is of the form $\mathbf{s}=(s_1, s_2, \ldots, s_{n \cdot m-1}, s_{n \cdot m})$ and $g_1(\mathbf{s}), g_2(\mathbf{s}), \ldots, g_{n-1}(\mathbf{s}), g_n(\mathbf{s})$ denote the n real-valued cost functions. The image of the vector-valued function $\mathbf{g}$ is the set $Z=\mathrm{im}(\mathbf{g})$. The image of a decision vector $\mathbf{s}\in Y$ is the n-dimensional vector $\mathbf{g}(\mathbf{s})\in Z$. To characterise mathematically the set of Pareto-optimal solutions of the multi-objective minimisation problem (4.6), it is necessary to extend the relational operators $=$, $\leq$ and $<$ to the elements of the set Z ([97], [99]). For any two vectors $\mathbf{u}, \mathbf{v} \in Z$,

$$\begin{aligned} &\mathbf{u} = \mathbf{v} \;\; \textit{iff} \;\; \forall\, i \in \{1, 2, \ldots, n-1, n\}: u_i = v_i \\ &\mathbf{u} \leq \mathbf{v} \;\; \textit{iff} \;\; \forall\, i \in \{1, 2, \ldots, n-1, n\}: u_i \leq v_i \\ &\mathbf{u} < \mathbf{v} \;\; \textit{iff} \;\; \mathbf{u} \leq \mathbf{v} \text{ and } \mathbf{u} \neq \mathbf{v} \end{aligned} \tag{4.7}$$

Considering (4.7), a candidate solution $\mathbf{s}^* \in Y$ is said to be Pareto-optimal if there is no $\mathbf{s}\in Y$ such that $\mathbf{g}(\mathbf{s}) < \mathbf{g}(\mathbf{s}^*)$. Hence, given the values of the costs $g_1(\mathbf{s}^*), g_2(\mathbf{s}^*), \ldots, g_{n-1}(\mathbf{s}^*), g_n(\mathbf{s}^*)$, no candidate solution $\mathbf{s}\in Y$ can cause a reduction in one of these costs without simultaneously causing an increase in at least one of the others. If $\mathbf{s}^1, \mathbf{s}^2 \in Y$ and $\mathbf{g}(\mathbf{s}^1) < \mathbf{g}(\mathbf{s}^2)$ then it is said that $\mathbf{s}^1$ dominates $\mathbf{s}^2$. The set of all the Pareto-optimal solutions is called the *Pareto set* and is denoted as Y^*. The Pareto set is considered the solution to the problem (4.6).

Most classical methods of generating the Pareto-optimal set are based on combining the n cost functions into a single, parameterised one ([96], [97]). Among those methods, one of the most commonly used is the *Linear Weighting Method*, which consists of converting the multi-objective optimisation problem to the minimisation of a linear combination of the cost functions. The new single-objective optimisation problem is expressed as follows:

$$minimise \;\; w(s) = w_1 g_1(s) + w_2 g_2(s) + \ldots + w_{n-1} g_{n-1}(s) + w_n g_n(s) \qquad (4.8)$$
$$subject\ to \;\; s \in Y$$

The factors w_i, with i=1, 2,..., n-1, n, are called *weights* and are assumed to be positive and normalised so that $\sum_{i=1}^{n} w_i = 1$. Given a fixed combination of values for the weights w_i, the resolution of the corresponding single-objective minimisation problem results in a Pareto-optimal solution of the original multi-objective optimisation problem. This can be easily proven by contradiction. Assume that a candidate decision vector $\boldsymbol{a} \in Y$ minimises $w(\boldsymbol{s})$ for a given combination of weights and that $\boldsymbol{a}$ is not Pareto-optimal. Then, there exists a candidate solution $\boldsymbol{b} \in Y$ such that $\boldsymbol{b}$ dominates $\boldsymbol{a}$. Suppose, without loss of generality, that $g_1(\boldsymbol{b}) < g_1(\boldsymbol{a})$ and $g_i(\boldsymbol{b}) \leq g_i(\boldsymbol{a})$ for i=2, 3,..., n-1, n. Therefore, according to the definition of the function $w(\boldsymbol{s})$ in (4.8), $w(\boldsymbol{b}) < w(\boldsymbol{a})$, which contradicts the assumption that $\boldsymbol{a}$ minimises $w(\boldsymbol{s})$.

4.4.1.2 Planning as searching for a Pareto-optimal solution to a multi-objective optimisation problem

The search for an allowable joint plan that results in an equitable distribution of the resolution costs among the team members is recast below as the search for a Pareto-optimal solution of a multi-objective optimisation problem. Considering the mathematical framework for the planning process introduced above, the following multi-objective minimisation problem can be formulated:

$$minimise \;\; \boldsymbol{C}^X(\boldsymbol{x}) = (C_1^X(\boldsymbol{x}),\ C_2^X(\boldsymbol{x}), \ldots,\ C_{n-1}^X(\boldsymbol{x}),\ C_n^X(\boldsymbol{x})) \qquad (4.9)$$
$$subject\ to \;\; \boldsymbol{x} \in D,\ D \subset X$$

The set D of candidate plans is formed by the plans belonging to the set X that are predicted to result in the resolution of all the conflicts in which the n team members are involved. Implicitly, this implies the introduction of a constraint restricting the candidate plans to those allowable ones that are anticipated to ensure that the distance between any pair of team members is kept above the established separation minima.

This constraint, which is denoted by R_D, cannot be expressed in a straightforward manner as a set of equality and/or inequality restrictions on certain explicit functions defined over the set X.

A Pareto-optimal solution to the problem (4.9) is a plan $x^* \in D$ such that no other candidate plan dominates x^*. Thus, there exists no allowable conflict-free plan that reduces the cost of the resolution trajectory of a team member without increasing the cost of the resolution trajectory of at least one other team member. Hence, a Pareto-optimal plan can be seen as resulting in the team members sharing the total conflict resolution costs, as it represents a trade-off among all the individual costs. A plan that does not belong to the Pareto set, denoted by D^*, would privilege some aircraft-agents over others. In principle, any $x^* \in D^*$ can be deemed as an acceptable outcome of the planning process. Thus, there is no need to search for the complete mathematical solution of (4.9), which would entail the generation of the entire Pareto set. This allows for the design of a planning algorithm whose goal is to generate a single Pareto-optimal solution to the multi-objective optimisation problem in (4.9).

4.4.2 Description of the proposed planning algorithm

The planning process has been recast as the search for a Pareto-optimal solution to the multi-objective optimisation problem in (4.9). By applying the Linear Weighting Method, which was described above, the generation of Pareto-optimal solutions to (4.9) is made equivalent to the resolution of the following single-objective minimisation problem:

$$\textit{minimise}\ \ w(x) = w_1 C_1^X(x) + w_2 C_2^X(x) + \ldots + w_{n-1} C_{n-1}^X(x) + w_n C_n^X(x) \qquad (4.10)$$

$$\textit{subject to}\ \ x \in D,\ D \subset X$$

For each combination of positive weights $w_1, w_2, \ldots, w_{n-1}, w_n$ such that $\sum_{i=1}^{n} w_i = 1$, the plan x^* that minimises (4.10) is a Pareto-optimal solution of (4.9).

The proposed planning algorithm is in fact an optimisation algorithm that searches for a plan that minimises (4.10) for a given combination of weights. As mentioned above, the set D is defined by the constraint R_D on the allowable plans, which restricts the candidate plans to those that are conflict-free. The difficulty to express the constraint R_D as a set of equality and/or inequality restrictions on functions of the variable x hinders the application of many classical optimisation methodologies to solve the problem (4.10). Consequently, some assumptions about the structure of the set D have been made with the objective of facilitating the optimisation process. The optimisation algorithm proposed implements an optimisation methodology that takes advantage of these assumptions, which are discussed below.

4.4.2.1 Assumptions about the structure of the set of candidate plans

As it has been explained above, when building the set of allowable resolution trajectories for a team member A_i, RT_i^A, the team organiser considers only the resolution trajectories that it believes are feasible for A_i. Among those resolution trajectories, the team organiser selects the ones that do not cause A_i to deviate excessively from its initially intended route. The selected resolution trajectories for A_i are expressed according to the mapping $L: RT_i^A \rightarrow X_i$, where X_i is a subset of $\mathbb{R}^m$. Each of the sets X_i contains the vectors defining the resolution trajectories deemed as allowable for A_i by the team organiser. It is assumed that the sets of allowable resolution trajectories of the team members contain a finite number of elements and, as a consequence, the sets X_i, for i=1, 2,..., n-1, n, are discrete sets. The number of allowable plans, which are formed by combining allowable resolution trajectories, is also finite and the set X of the allowable plans is a discrete set. Consequently, the set D is also a discrete set, inasmuch as it is a subset of X.

As it was done for the trajectory-planning algorithm presented in the previous chapter, the concept of conflict resolution patterns is also introduced here to simplify the optimisation process. The team organiser groups the resolution trajectories in each of the sets RT_i^A, for i=1, 2,..., n-1, n, into mutually exclusive subsets, containing those resolution trajectories that belong to the same morphological category. It is assumed that there are a finite number, n_p, of such categories, which are common to all the team members. The conflict resolution patterns are specific pre-defined resolution trajectories that represent each of those subsets for a given team member. These conflict resolution patterns are elements of X_i and are denoted as x_i^1, x_i^2,..., $x_i^{n_p-1}$, $x_i^{n_p}$.

The concept of conflict resolution patterns can be applied in the context of the allowable plans by introducing the concept of *plan-pattern*. A plan-pattern is defined as an allowable plan of the form $x^{i_p}=(x_1^{j_1}, x_2^{j_2},..., x_{n-1}^{j_{n-1}}, x_n^{j_n})$, where $i_p \in \{1, 2,..., (n_p)^n-1, (n_p)^n\}$ and $j_1, j_2,..., j_{n-1}, j_n \in \{1, 2,..., n_p-1, n_p\}$. The plan-pattern x^{i_p} represents the category of allowable plans comprising of those formed by a combination of resolution trajectories such that A_i's resolution trajectory within the plan fits the conflict resolution pattern $x_i^{j_i}$, for i= 1, 2,..., n-1, n. Considering that each of the n team members can be assigned n_p different conflict resolution patterns, the number of plan-patterns, which is denoted as n_{pp}, is given by the n-permutations with repetition of n_p elements. Hence, $n_{pp}=(n_p)^n$. The elements of the set of the allowable plans X can be classified into n_{pp} mutually exclusive subsets, each of which is represented by one of the plan-patterns. These subsets contain allowable plans that can be classified into the same morphological category. Consequently, each of the elements of the set of candidate plans D, which is a subset of X, belongs to one of those n_{pp} subsets and is said to fit the corresponding plan-pattern.

4.4.2.2 Metaheuristic techniques for combinatorial optimisation

Considering the assumptions introduced above, the problem (4.10) is defined over a discrete set and, consequently, it must be treated as a combinatorial optimisation problem. Combinatorial optimisation problems arise in situations where an optimal solution to a given objective must be found among a finite or countably infinite number of alternatives [103]. Near-optimal solutions to combinatorial optimisation problems can be found within reasonable computation times using metaheuristic techniques ([100], [102]). The term metaheuristic derives from the composition of two words of Greek origin, the prefix "meta", which implies "beyond" or "of a higher form", and "heuristic", derived from verb "heuriskein", which means "to find" or "to discover". In the context of combinatorial optimisation, heuristics are techniques used to guide the search for reasonably good solutions to a complicated problem at a moderate computational cost. A heuristic consists of a rule or a set of rules generally based on intuition. The design of heuristics usually involves using knowledge about the structure of the specific problem at hand to devise a strategy to find an approximate solution to it ([100], [42]). Considering the foregoing, the term metaheuristic is generally used to refer to high-level optimisation strategies that can be used to direct the search for near-optimal solutions without requiring a strong insight into the structure of the specific problem being solved [102]. A metaheuristic can be seen as a general-purpose heuristic applicable to different combinatorial optimisation problems [99]. Metaheuristics may also operate as master processes that guide and modify subordinate problem-specific heuristics to efficiently produce high-quality solutions [100]. The following combinatorial optimisation algorithms, among others, can be classified as metaheuristic techniques:

- Iterated Local Search [104].
- Simulated Annealing [105].
- Tabu Search [106].
- Ant Colony Optimisation [107].
- Genetic Algorithms [46].

4.4.2.3 Proposed optimisation algorithm: *modified multi-start random mutation hillclimbing*

The optimisation algorithm proposed to solve the problem (4.10) is based on one of the simplest metaheuristic techniques: *random mutation hillclimbing* [99]. Before describing the algorithm, it is necessary to introduce the concept of *neighbourhood* [103], which is the cornerstone of random mutation hillclimbing. In the context of the problem (4.10), a neighbourhood function is a mapping N: $D \rightarrow 2^D$ that defines for each solution $x \in D$ a set $N(x) \subseteq D$ of solutions that are in some sense close to x. The set $N(x)$ is the neighbourhood of the solution x and each $y \in N(x)$ is said to be a neighbour of x. Considering the assumptions introduced above about the structure of the set D, a neighbourhood function is defined so that the set $N(x)$ coincides with the set of the candidate plans that fit the same plan-pattern as x. Thus, if the candidate plan x fits the plan-pattern x^{i_p}, the set $N(x)$ contains all the candidate plans y that fit x^{i_p}.

The operation of the algorithm begins by generating an initial candidate plan $x_{init} \in D$ and evaluating the value of $w(x_{init})$. The initial candidate plan x_{init} becomes the *current candidate plan*, which is denoted by x_c. Then, the algorithm performs an iterative improvement process, at each step of which a new candidate plan x_r is selected at random among the neighbours of x_c (a *random mutation* is applied to the current candidate solution). If $w(x_r) > w(x_c)$, then x_r is discarded. On the other hand, if $w(x_r) \leq w(x_c)$, then x_r becomes the current candidate plan, $x_c = x_r$. The iterative improvement process is performed during a pre-established time span of t_{it}. The output of the iterative process is the current candidate solution at the time when the process is halted.

If the iterative improvement described above were allowed to run indefinitely, then the current candidate plan would converge to a local minimum of the problem (4.10) [99]. A candidate plan x_{lm} is said to be a local minimum with respect to N if $w(x_{lm}) \leq w(y)$, for all $y \in N(x_{lm})$ [103]. Thus, assuming that the time t_{it} is sufficiently long, the output of the iterative improvement process can be considered as a satisfactory approximation to a local minimum of the problem (4.10). However, a local minimum x_{lm} may not be the solution to the problem (4.10). That is, there may be candidate plans in D that do not belong to $N(x_{lm})$ but that result in a value of w lower than $w(x_{lm})$. To alleviate this problem, the iterative improvement process described above is carried out several times re-starting at different initial candidate solutions. This technique, which is frequently used to avoid local optima, is called *multi-start random mutation hillclimbing* ([99], [101]). The application of the multi-start random mutation hillclimbing metaheuristic requires re-starting the iterative improvement process several times with a random initial solution. The quality of the solution obtained depends on the number of re-stars. If the iterative improvement process were allowed to re-start with a random initial candidate plan an infinite number of times, the resulting hypothetical algorithm would produce an approximation to the global minimum of (4.10).

The multi-start random mutation hillclimbing metaheuristic has been adapted to the specific problem considered here to take advantage of the structure of the set D of candidate plans. Instead of using random re-starts, the iterative improvement process is successively re-started with an initial candidate solution given by a plan-pattern that belongs to D. Thus, the algorithm performs as many iterative improvement processes as the number of conflict-free plan-patterns. After all the prescribed re-starts have been performed, the algorithm outputs the plan with the lowest value of the function w among those that result from the successive iterative improvement processes.

The algorithm introduced above, which is described in Figure 4.6 using pseudocode, has been named *modified multi-start random mutation hillclimbing* (M-MRMH). The plan output by the algorithm, which is denoted by $\tilde{x}^*$, is deemed to be a valid approximation of the minimum of (4.10), as it is the best local minimum among those obtained searching the neighbourhoods of the conflict-free plan-patterns. Consequently, the plan $\tilde{x}^*$ can be regarded as an approximation of an element of the Pareto set of the problem (4.9). The algorithm is guaranteed to produce a plan $\tilde{x}^*$ within a finite time limit in the order of $r \cdot t_{it}$, where r is the number of conflict-free plan-patterns. The plan obtained is neither unique nor predictable *a priori* due to the random elements in the algorithm. Thus, if the algorithm were run several times for the same minimisation problem, the $\tilde{x}^*$'s obtained would vary randomly.

The M-MRMH algorithm is proposed as a planning algorithm suitable to be used when applying the reflective co-operation mechanism. The remainder of this chapter is devoted to illustrate the performance of the mechanism in two-dimensional conflict scenarios assuming that the team organiser elaborates the joint conflict resolution plans using the M-MRMH algorithm.

4.5 Application of the reflective co-operation mechanism in two-dimensional conflicts

The M-MRMH algorithm described above has been implemented and adapted to conflict scenarios involving up to three Autonomous Aircraft cruising at the same

Algorithm ModifiedMulti-startRandomMutationHillclimbing

Let r be the number of conflict-free plan-patterns, with $r \leq (n_p)^n$

Let $x^1, x^2, \ldots, x^{r-1}, x^r \in D$ be the r conflict-free plan-patterns

Let $N: D \rightarrow 2^D$ be the neighbourhood function defined as:

$N(x) = \{y \in D \mid y$ fits the same plan-pattern as $x\}$

Let $w_1, w_2, \ldots, w_{n-1}, w_n$ be n given arbitrary positive weights such that $\sum_{i=1}^{n} w_i = 1$

Let $w(x) = w_1\, C_1^X(x) + w_2\, C_2^X(x) + \ldots + w_{n-1}\, C_{n-1}^X(x) + w_n\, C_n^X(x)$ be the cost function

Let t_{it} be the duration of the iterative improvement process

For each i from 1 up to r **do**

$c^i = x^i$

$w_c^i = w(x^i)$

t_0 = Current_Time

$ElapsedTime = 0$

While $ElapsedTime \leq t_{it}$ **do**

Let N_i be the neighbourhood of x^i, which is given by $N(x^i)$

Generate a random $x^{rand} \in N_i$

if $w(x^{rand}) \leq w_c^i$

$c^i = x^{rand}$

$w_c^i = w(x^{rand})$

$ElapsedTime$ = Current_Time-t_0

Let x^* be c^i with $i \in \{1, 2, \ldots, r-1, r\}$ such that $w_c^i = \min\{ w_c^1, w_c^2, \ldots, w_c^{r-1}, w_c^r \}$

Output x^* and $w(x^*)$

Figure 4.6. Pseudocode of the M-MRMH algorithm

altitude, with the objective of simulating the application of the reflective co-operation mechanism in such conflict scenarios. In this section, the performance of the reflective co-operation mechanism with the M-MRMH algorithm in these types of scenarios will be investigated and compared to that of the behaviouristic co-operation mechanism, which was analysed in the previous chapter. To allow for this comparison, a series of assumptions have been made regarding the process of conflict detection, the response latencies of the flight crews, the set of allowable conflict resolution plans, the plan-patterns and the cost functions of the aircraft-agents. These assumptions are discussed below.

4.5.1 Conflict detection

The aircraft-agents involved in a conflict scenario are assumed to be flying user-preferred routes at their respective optimal airspeeds. They continuously broadcast their position, speed and intentions through ADS-B. Based upon the received ADS-B information, each aircraft predicts the future trajectories of its proximate aircraft and compare them with its own intended trajectory to detect possible conflicts. The guidance capability of the 4D-FMS enables the aircraft to comply with their nominal four-dimensional trajectories within AAO airspace with a certain degree of accuracy. It is assumed that the intent information contained in the ADS-B messages is sufficiently complete and accurate to enable aircraft to reconstruct one another's intended four-dimensional trajectories within AAO airspace. Thus, as soon as two aircraft enter one another's ADS-B range of coverage, they are able to detect if a conflict is to occur between them within AAO airspace, irrespectively of the time to the Closet Point of Approach.

An aircraft-agent declares a conflict when it predicts that another aircraft's nominal intended trajectory will penetrate its Protection Volume. When an aircraft anticipates a violation of its Protection Volume at a certain four-dimensional point along its intended trajectory within AAO airspace, the probability of a violation of its Collision Volume at that point is higher than the Target Level of Safety and a conflict resolution action must be taken. The dimensions of the Protected Volume reflect the guidance precision of the aircraft's 4D-FMS and the accuracy of the intent information transmitted via ADS-B. As in the previous chapter, it is assumed here that the Protection Volume in the two-dimensional conflicts considered consists of a circle of radius 5 nautical miles centred on the aircraft, irrespectively of the look-ahead time for conflict detection. This 5 nautical miles separation minimum has been adopted inasmuch as it is currently used as a standard for radar separation. No errors are assumed to occur in the conflict detection process and false alarms are not considered.

4.5.2 Model of the flight crew response latency

The application of the co-operation mechanism requires the flight crews to become aware of the conflicts in which they are involved and to understand and approve the formation of a team to resolve those conflicts. In the example of team formation process presented in section 4.3.2, it was assumed that the flight crew acceptance latency to a request for permission to join a team is a random time uniformly distributed between 40 seconds and 80 seconds. Thus, the PDF of the flight crew response latency in that case was the same as the one used to model the flight crew response latency in the previous chapter, which was depicted in Figure 4.3. The example showed that the team formation process is further delayed by the fact that the aircraft-agents have to consider all the team membership proposals that they either receive or issue during a time span, t_f, before deciding which team to join. In the example the value of t_f was 30 seconds. According to the assumptions made in this example, the team formation process in a generic conflict scenario can take up to 110 seconds, measured from the time a team proposal comprising of all the aircraft in the conflict scenario is issued.

Once a team has been formed, the team organiser applies the M-MRMH algorithm to elaborate a joint plan and subsequently transmits to each of its team members its respective resolution trajectory within the joint plan. Upon receiving its resolution trajectory, the aircraft-agent makes its flight crew aware of it. It is assumed that, in nominal operation, the resolution trajectories within the plan result in the resolution of all the conflicts in which the team members are involved. Consequently, each flight crew accepts the implementation of its respective resolution trajectory after a certain time lag. In principle, it could be assumed that this response latency is also modelled by the uniform distribution in Figure 4.3. Thus, the joint plan should not require any team member to deviate from its initially intended until at least 80 seconds after the team is formed to ensure that all the flight crews have enough time to understand and accept their respective resolution trajectories.

Considering the discussion above, the following assumptions regarding the value of t_f and the response latencies of the flight crews are made with the objective of simplifying the application of the co-operation mechanism in the conflict scenarios investigated in this section:

- The selected value of t_f is assumed to ensure that the team formation process always results in a team encompassing all the aircraft in the scenario.
- The response latencies of the flight crews are assumed to be such that all the resolution trajectories in the joint conflict resolution plan are accepted by their respective flight crews within 200 seconds of the time the team membership proposal encompassing all the aircraft in the scenario is issued.

The team membership proposal encompassing all the aircraft in the scenario is issued as soon as one of the aircraft is within ADS-B range of coverage of all the others. The initial positions of the aircraft in the conflict scenarios considered in this chapter correspond to the time when that team membership proposal is issued.

The second assumption above is based on the model of the flight crew response latency used in the previous chapter and on a value of t_f of 30 seconds. Considering that the flight crews can take up to 80 seconds to accept joining the team and other up to 80 seconds to approve their respective resolution trajectories, a 200 seconds time limit to establish and approve the joint plan provides a minimum of 10 seconds of buffer time for communication, data processing and computation of the joint plan. This assumption has been expressed in terms of a time limit for the acceptance of the joint plan so that it may be re-formulated easily for different values of t_f and other models of the flight crew response latency.

4.5.3 Set of allowable conflict resolution plans

For simplicity and to allow for the comparison between the performance of the two co-operation mechanisms in two-dimensional conflicts, it is assumed that the aircraft's resolution trajectories are restricted to the type of lateral shift manoeuvres described in the previous chapter. These manoeuvres are as shown in Figure 4.7. Given an aircraft's initial four-dimensional position, WP_0, and its initial heading, ψ_0, a lateral shift manoeuvre is unambiguously defined by the values of the three parameters γ, t_s and t_e.

The team organiser uses these three parameters as the components of a vector that univocally defines a resolution trajectory. This vector can be seen as a way of expressing the resolution trajectory in a hypothetical trajectory language.

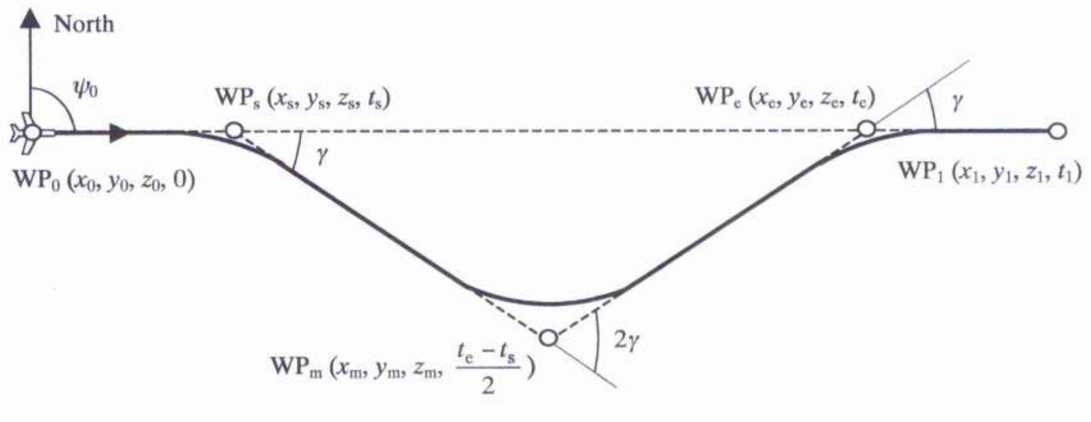


Figure 4.7: Generic lateral shift manoeuvre.

Considering the foregoing, a joint conflict resolution plan for a team comprised of n aircraft-agents would be expressed as the following vector of dimension $3 \cdot n$:

$$\boldsymbol{x}=(\gamma^{1}, t_s^1, t_e^1, \gamma^{2}, t_s^2, t_e^2, \ldots, \gamma^{n-1}, t_s^{n-1}, t_e^{n-1}, \gamma^{n}, t_s^n, t_e^n) \quad (4.11)$$

The 3-dimensional vector $\boldsymbol{x}_i=(\gamma^{i}, t_s^i, t_e^i)$ with $i \in \{1, 2,\ldots, n-1, n\}$ represents the resolution trajectory assigned to the team member A_i in the joint plan $\boldsymbol{x}$. The times t_s^i and t_e^i are measured from the time at the initial waypoint of the corresponding resolution trajectory, which is denoted by WP_0^i. The team organiser sets the time when it sent the team membership proposal message to its n-1 team mates as the common time origin for the n resolution trajectories that form the plan. Hence, the waypoints WP_0^i for i=1, 2,…, n-1, n correspond to the team members' positions at that time origin.

4.5.3.1 Conflict resolution patterns and plan-patterns

The resolution trajectories allowable to the team members are classified into three mutually exclusive groups. Each of these groups comprises of those resolution trajectories that are considered as belonging to the same category defined by certain common geometrical characteristics. For each team member, one resolution trajectory of each group is selected as the conflict resolution pattern representing that group. The three conflict resolution patterns are schematically depicted in Figure 4.8. Pattern 1 coincides with the initially intended trajectory and can be seen as a lateral shift manoeuvre with $\gamma = 0^{o}$. Patterns 2 and 3 represent left and right lateral shift manoeuvres, respectively. For each team member A_i, these two patterns are two symmetrical lateral shift manoeuvres determined by the three positive parameters $\gamma^{p,i}$, $t_s^{p,i}$ and $t_e^{p,i}$. Consequently, the three conflict resolution patterns for A_i are given by $\boldsymbol{x}_i^1$=(0,0,0), $\boldsymbol{x}_i^2 = (-\gamma^{p,i}, t_s^{p,i}, t_e^{p,i})$ and $\boldsymbol{x}_i^2 = (\gamma^{p,i}, t_s^{p,i}, t_e^{p,i})$.

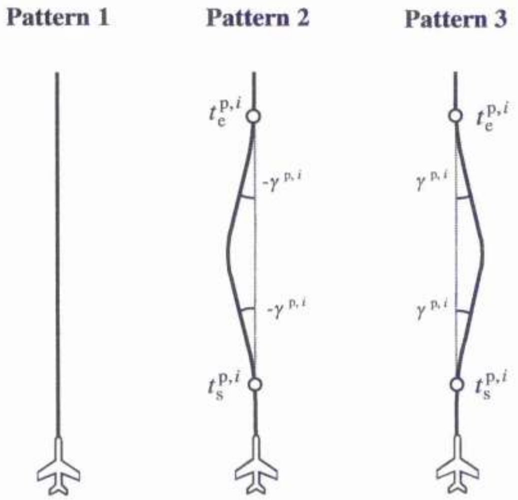


Figure 4.8. Conflict resolution patterns for the team member A_i

The above categorisation of the resolution trajectories allowable to the team members induces a categorisation among the allowable conflict resolution plans. Considering the three groups of allowable resolution trajectories introduced above, the allowable plans are classified into 3^n groups. Each of these groups is represented by a plan-pattern that assigns to each team member one of its three conflict resolution patterns. Thus, a plan-pattern is defined as $\boldsymbol{x}^{i_p} = (\boldsymbol{x}_1^{j_1}, \boldsymbol{x}_2^{j_2}, \ldots, \boldsymbol{x}_{n-1}^{j_{n-1}}, \boldsymbol{x}_n^{j_n})$, where $j_1, j_2, \ldots, j_{n-1}, j_n \in \{1, 2, 3\}$ and $i_p \in \{1, 2, \ldots, 3^n\text{-}1, 3^n\}$. For example, in a team formed by two aircraft-agents A_1 and A_2, the 9 plan-patterns would be defined as follows:

$$
\begin{aligned}
x^1 &= (x_1^1, x_2^1) = (0, 0, 0, 0, 0, 0) \\
x^2 &= (x_1^2, x_2^2) = (-\gamma^{p,1}, t_s^{p,1}, t_e^{p,1}, -\gamma^{p,2}, t_s^{p,2}, t_e^{p,2}) \\
x^3 &= (x_1^3, x_2^3) = (\gamma^{p,1}, t_s^{p,1}, t_e^{p,1}, \gamma^{p,2}, t_s^{p,2}, t_e^{p,2}) \\
x^4 &= (x_1^1, x_2^2) = (0,0,0,-\gamma^{p,2}, t_s^{p,2}, t_e^{p,2}) \\
x^5 &= (x_1^2, x_2^1) = (-\gamma^{p,1}, t_s^{p,1}, t_e^{p,1}, 0,0,0) \\
x^6 &= (x_1^1, x_2^3) = (0,0,0,\gamma^{p,2}, t_s^{p,2}, t_e^{p,2}) \\
x^7 &= (x_1^3, x_2^1) = (\gamma^{p,1}, t_s^{p,1}, t_e^{p,1}, 0,0,0) \\
x^8 &= (x_1^2, x_2^3) = (-\gamma^{p,1}, t_s^{p,1}, t_e^{p,1}, \gamma^{p,2}, t_s^{p,2}, t_e^{p,2}) \\
x^9 &= (x_1^3, x_2^2) = (\gamma^{p,1}, t_s^{p,1}, t_e^{p,1}, -\gamma^{p,2}, t_s^{p,2}, t_e^{p,2})
\end{aligned}
\tag{4.12}
$$

The plans belonging to the group defined by a given plan-pattern are formed by resolution trajectories that fit the corresponding conflict resolution pattern for each team member. For example, in the case of a team with two members and according to the definition of the plan-patterns in (4.12), the plans fitting the plan-pattern x^6 assign to A_1 a resolution trajectory that fits A_1's Pattern 1 and to A_2 a resolution trajectory that fits A_2's Pattern 3.

4.5.3.2 Values assigned to the parameters $\gamma^{p,i}$, $t_s^{p,i}$ and $t_e^{p,i}$

It is assumed that the team organiser sets the same value of $\gamma^{p,i}$ for all the team members, and hence $\gamma^{p,i} = \gamma^p$ for i =1, 2,..., n-1, n. As it was done when defining the conflict resolution patterns in the previous chapter, γ^p is initially set to 10°. If this value results in no conflict-free plan-patterns, then γ_p is successively incremented by 5° until at least one conflict-free pattern is found. The value of the time $t_s^{p,i}$ is also assumed to be the same for all the team members, $t_s^{p,i} = t_s^p$ for i =1, 2,..., n-1, n. The value of t_s^p is set to 200 seconds, which coincides with the time by which the joint plan must have been approved by the flight crews of all the team members.

The value of the time $t_e^{p,i}$, which may differ from one team member to another, is defined as a function of the average of the times of the Closest Points of Approach in all the conflicts in which the corresponding team member is involved. This average time is denoted by $t^{av}_{\mathrm{CPA},i}$ and $t^{\mathrm{p},i}_{\mathrm{e}}$ is given by:

$$t^{\mathrm{p},i}_{\mathrm{e}} = t^{\mathrm{av}}_{\mathrm{CPA},i} + (t^{\mathrm{av}}_{\mathrm{CPA},i} - t^{\mathrm{p}}_{\mathrm{s}}) \tag{4.13}$$

4.5.3.3 Definition of the set of allowable conflict resolution plans

The set X_i of allowable resolution trajectories for the team member A_i can be expressed in terms of the Cartesian product $\Gamma_i \times \mathrm{T}^i_s \times \mathrm{T}^i_e$, where Γ_i, T^i_s and T^i_e, with $i \in \{1, 2, \ldots, n\text{-}1, n\}$, denote, respectively, the sets of allowable values of γ^i, t^i_s and t^i_e. The sets Γ_i for i=1, 2,..., n-1, n, are all assumed to be equal to a set denoted by Γ and defined as follows:

$$\Gamma = \{-\gamma^{\mathrm{P}}, -\gamma^{\mathrm{P}}+1, \ldots, -1, 0, 1, \ldots, \gamma^{\mathrm{P}}-1, \gamma^{\mathrm{P}}\} \tag{4.14}$$

where the elements of Γ are expressed in degrees. The angles γ^i<0 corresponds to resolution trajectories fitting Pattern 1. The angles $\gamma^i \in \Gamma$ such that γ^i<0 correspond to resolution trajectories fitting Pattern 2. The angles $\gamma^i \in \Gamma$ such that γ^i>0 correspond to resolution trajectories fitting Pattern 3.

Since it is assumed that no resolution trajectory requires the aircraft to start deviating from its initially intended trajectory within 200 seconds of the initial time, the value of $t^{\mathrm{p}}_{\mathrm{s}}$ is set to 200 seconds. The set T^i_s is defined as follows:

$$\mathrm{T}^i_s = \{ t^{\mathrm{p}}_{\mathrm{s}},\ t^{\mathrm{p}}_{\mathrm{s}}+1,\ t^{\mathrm{p}}_{\mathrm{s}}+2, \ldots,\ t^{av}_{\mathrm{CPA},i}\text{-}2,\ t^{av}_{\mathrm{CPA},i}\text{-}1,\ t^{av}_{\mathrm{CPA},i} \} \tag{4.15}$$

where elements of T^i_s are expressed in seconds.

The set T^i_e is defined as follows:

$$\mathrm{T}^i_e = \{ t^{av}_{\mathrm{CPA},i},\ t^{av}_{\mathrm{CPA},i}+1,\ t^{av}_{\mathrm{CPA},i}+2, \ldots,\ t^{\mathrm{p},i}_{\mathrm{e}}\text{-}2,\ t^{\mathrm{p},i}_{\mathrm{e}}\text{-}1,\ t^{\mathrm{p},i}_{\mathrm{e}} \} \tag{4.16}$$

where elements of T_s^i are expressed in seconds.

A restriction has been imposed on the elements of the set X_i to allow for sufficient time between the first and the last turn so that the resolution trajectory is feasible. This restriction establishes that the difference between t_e^i and t_s^i must be at least 120 seconds. Considering this restriction, the set X_i is defined as follows:

$$X_i = \{x_i \in \Gamma_i \times \mathrm{T}_s^i \times \mathrm{T}_e^i \mid t_e^i - t_s^i \geq 120\} \tag{4.17}$$

The set X of allowable plans is defined as the following Cartesian product:

$$X = X_1 \times X_2 \times \ldots \times X_{n-1} \times X_n \tag{4.18}$$

where the X_i for $i = 1, 2, \ldots, n-1, n$ are given by (4.17). The plans belonging to X that are conflict-free form the set of candidate plans D.

4.5.4 Cost function

With the objective of comparing the costs resulting from the application of the two co-operation mechanism in a given conflict scenario, the type of cost function considered in this chapter is the same as in the previous chapter. To allow for a comparison of the costs among the team members, it is assumed that all of them apply the same cost-efficiency criteria. The cost function of the team member A_i is given by:

$$C_i(\gamma^i, t_s^i, t_e^i) = w_D D(\gamma^i, t_s^i, t_e^i) + w_M M(\gamma^i) \tag{4.19}$$

where $w_D = w_M = 1$ and the functions $D(\gamma, t_s, t_e)$ and $M(\gamma)$ were described in section 3.4.4.

Considering (4.19), the multi-objective optimisation problem in (4.9) takes the form:

$$\text{minimise } C^X(x)=(C_1(\gamma^1,t_s^1,t_e^1),\, C_2(\gamma^2,t_s^2,t_e^2),\ldots \tag{4.20}$$
$$\ldots,\, C_{n-1}(\gamma^{n-1},t_s^{n-1},t_e^{n-1}),\, C_n(\gamma^n,t_s^n,t_e^n))$$
$$\text{subject to } x\in D$$

where $x=(\gamma^1,\ t_s^1,\ t_e^1,\ \gamma^2,\ t_s^2,\ t_e^2,\ \ldots,\ \gamma^{n-1},\ t_s^{n-1},\ t_e^{n-1},\ \gamma^n,\ t_s^n,\ t_e^n)$. To find an approximation of an element of the Pareto set of (4.20), the M-MRMH algorithm will be applied to the following single-objective optimisation problem:

$$\text{minimise } w(x) = w_1C_1(\gamma^1,t_s^1,t_e^1) + w_2C_2(\gamma^2,t_s^2,t_e^2) + \ldots$$
$$\ldots + w_{n-1}C_{n-1}(\gamma^{n-1},t_s^{n-1},t_e^{n-1}) + w_nC_n(\gamma^n,t_s^n,t_e^n) \tag{4.21}$$
$$\text{subject to } x\in D$$

In principle, the values of the weights w_i, with i=1, 2,..., n-1, n, can be chosen arbitrarily as long as they are positive and normalised so that $\sum_{i=1}^{n} w_i = 1$.

4.5.5 Two-dimensional conflicts between two aircraft

This section illustrates the application of the reflective co-operation mechanism with the M-MRMH planning algorithm in two-dimensional conflict scenarios involving two aircraft. With the objective of comparing the performance of this co-operation mechanism with the behaviouristic one, the two-aircraft conflict scenarios investigated in this section are the same as those for which the performance of the behaviouristic mechanism was analysed in the previous chapter.

4.5.5.1 Simulation of the application of the reflective co-operation mechanism

The application of the reflective co-operation mechanism in a conflict scenario involving two aircraft results in one of the two aircraft acting as the team organiser and forming a team encompassing both of them. In this type of scenarios, only one team membership proposal is issued. Once the team has been established, the team organiser runs the M-MRMH algorithm to elaborate a joint plan and, subsequently, it presents its flight crew with its assigned resolution trajectory and transmits to the other team member its corresponding one. The two flight crews accept their respective resolution trajectories within 200 seconds of the time when the team organiser sends the team membership proposal message to the other aircraft. This time coincides with the time when the two aircraft enter each other's ADS-B range of coverage, which corresponds to the initial positions of the aircraft in the conflict scenario. As mentioned above, the resolution trajectories produced by the M-MRMH algorithm do not require the aircraft to alter their routes within 200 seconds of the initial time. Hence, the resolution trajectories that result from the application of the co-operation mechanism are independent of the response times of the flight crews. The resolution trajectories are also independent of which of the two aircraft acts as the team organiser, since both aircraft use the M-MRMH to elaborate the joint plan.

Considering the assumptions discussed in section 4.4.4, the M-MRMH algorithm has been adapted specifically to two-dimensional conflict scenarios involving two aircraft. The function $w(\boldsymbol{x})$ to be minimised by the algorithm in such conflicts is as follows:

$$w(\boldsymbol{x}) = \frac{1}{2} C_1(\gamma^1, t_s^1, t_e^1) + \frac{1}{2} C_2(\gamma^2, t_s^2, t_e^2) \qquad (4.21)$$

where both weights w_1 and w_2 have been set to $\frac{1}{2}$. These values of the weights comply with the requirements w_1, $w_2 > 0$ and $w_1 + w_2 = 1$. The minimum $\boldsymbol{x}^*$ of $w(\boldsymbol{x})$ is also the minimum of the function $C_t(\boldsymbol{x}) = 2w(\boldsymbol{x}) = C_1(\gamma^1, t_s^1, t_e^1) + C_2(\gamma^2, t_s^2, t_e^2)$, which is the total cost of resolving the conflict. Hence, besides belonging to the Pareto set of the multi-objective minimisation problem involving the two individual cost functions, the plan $\boldsymbol{x}^*$ results in the lowest possible total resolution cost.

The algorithm has been implemented using the MATLAB® computing language. To simulate the application of the co-operation mechanism, the resolution trajectories that result of running the algorithm are generated and visualised using the MATLAB® integrated computing environment. The algorithm is run using the version 5.2 of

MATLAB® for Windows® 95 on a desktop PC with 64 MB RAM equipped with an Intel® Pentium® MMX processor operating at 233 MHz CPU clockspeed. The version of MATLAB® and the computing platform used are the same as those used to run the simulations of the application of the behaviouristic mechanism presented in the previous chapter.

The CPU time limit for the iterative improvement processes performed by the M-MRMH algorithm, t_{it}, has been set to 10 seconds. In an airborne implementation of the algorithm, this CPU time limit might excessively lengthen the computation time to produce a joint plan and cause the acceptance of the resolution trajectories by the flight crews to be delayed beyond the 200 seconds limit. Considering the increasing computing power available on board modern aircraft, it is assumed that the value of the time limit t_{it} in a hypothetical airborne implementation of the M-MRMH algorithm could be reduced so that a joint plan can be approved by the flight crews before the 200 seconds limit.

The M-MRMH algorithm is based on the random exploration of neighbourhoods and, as a result, the output $\tilde{x}^*$ of the algorithm is also random. Hence, to analyse the performance of the co-operation mechanism in a given conflict scenario, the output of the algorithm for that scenario will be studied from a statistical perspective. As done for the behaviouristic mechanism, the four random variables $\mathbf{c_1}$, $\mathbf{c_2}$, $\mathbf{s_c}$ and $\mathbf{d_{A1\text{-}A2}}$ are defined to describe the performance of the reflective co-operation mechanism in a specific conflict scenario. These random variables denote, respectively, the cost of the resolution trajectories assigned to each of the two conflicting aircraft, the sum of these two costs and the predicted minimum distance between the aircraft as they fly their resolution trajectories. Each of the random variables has an associated probability density function (PDF). The PDFs of the random variables $\mathbf{c_1}$, $\mathbf{c_2}$, $\mathbf{s_c}$ and $\mathbf{d_{A1\text{-}A2}}$ are denoted as $f(c_1)$, $f(c_2)$, $f(s_c)$, and $f(d_{A1\text{-}A2})$, respectively. According to this notation, the PDF under consideration is identified by the independent variable of the function f. In this case, the PDFs above reflect only the probabilistic nature of the algorithm, as the response times of the flight crews do not influence the joint plans. This contrasts with the case of the behaviouristic mechanism, in which the response latencies of the flight crews heavily influence the result of applying the mechanism and, consequently, the

PDFs encode the randomness of those response latencies as well as the probabilistic nature of the trajectory-planning algorithm.

The statistical analysis of the PDFs $f(c_1)$, $f(c_2)$, $f(s_c)$, and $f(d_{A1\text{-}A2})$ for a given conflict scenario is based on repeatedly running the M-MRMH algorithm for that conflict scenario with the objective of obtaining a sample of each of the four random variables c_1, c_2, s_c and $d_{A1\text{-}A2}$. Providing that the sample size, which is given by the number of times the algorithm is run, is sufficiently large, the statistics of the samples obtained can be used to make inferences about the PDFs according to which those samples have been drawn. As in the previous chapter, the sample size has been chosen to be 50 so that the Central Limit Theorem holds and the results obtained with the two co-operation mechanism can be compared.

4.5.5.2 Application of the reflective co-operation mechanism in conflict scenario 1

This section will show the results of simulating the application of the mechanism in conflict scenario 1, which is reproduced in Figure 4.9. The application of the mechanism results in the formation of a team comprising both conflicting aircraft, one of which acts as the team organiser and elaborates a joint plan using the M-MRMH algorithm. This plan is independent of which of the two aircraft acts as the team organiser. The resolution trajectories that form the joint plan obtained from a run of the algorithm are shown in Figure 4.10 together with the predicted distances between the two aircraft as they fly those resolution trajectories. Table 4.2 displays the parameters defining the two resolution trajectories as well as the cost associated to each of them and their sum, which is regarded as the total cost of resolving the conflict. According to the joint plan obtained, A_1 is assigned a resolution trajectory that coincides with its initially intended route while A_2 is assigned a right lateral shift manoeuvre. In this case, A_2 bears all the cost of resolving the conflict, as it occurs when applying the behaviouristic mechanism assuming that A_2 acts as the leader.

To obtain samples of the random variables c_1, c_2, s_c and $d_{A1\text{-}A2}$ in this scenario, the M-MRMH algorithm has been run 50 times. The samples reflect the probabilistic character of the algorithm and are drawn according to the PDFs $f(c_1)$, $f(c_2)$, $f(s_c)$ and $f(d_{A1\text{-}A2})$, respectively. The relative frequency graphs of the samples obtained as well as

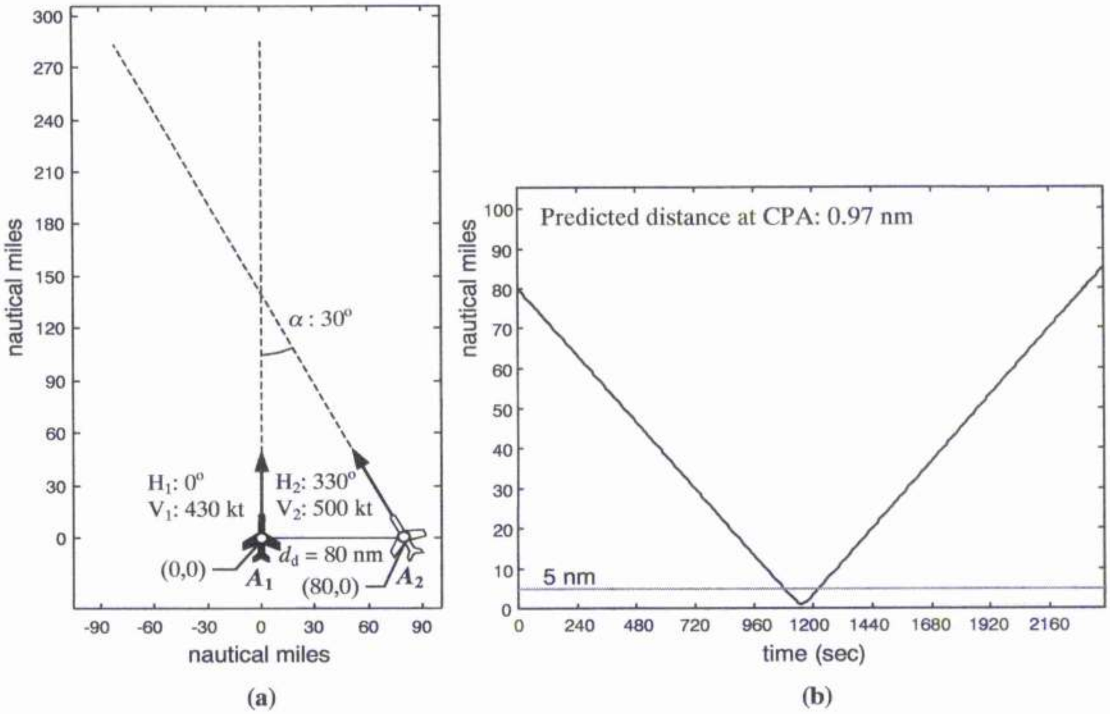


Figure 4.9. Conflict scenario 1. (a) Initial configuration (b) Predicted distances between the aircraft as they fly along their initially intended routes

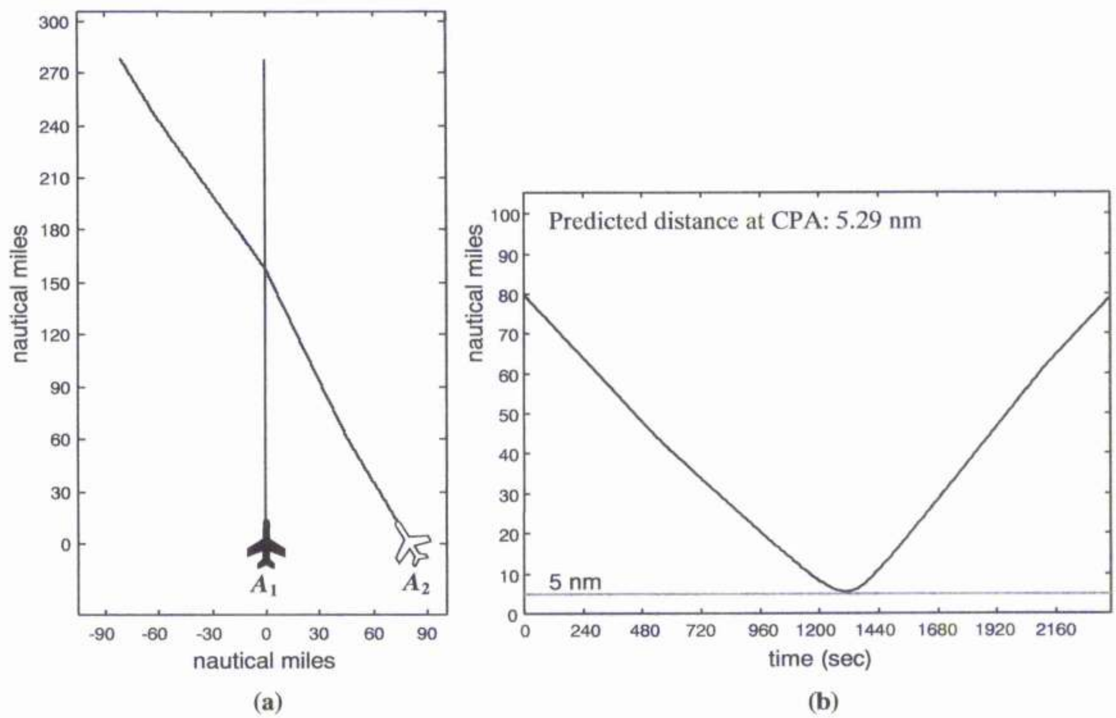


Figure 4.10. Simulation of the application of the M-MRMH planning algorithm in conflict scenario 1. The resulting resolution trajectories are independent of which of the two aircraft is the team organiser. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their assigned resolution trajectories

	γ	t_s	t_e	Cost
A_1	Initially intended route			0.00
A_2	5°	508 s	2044 s	2.06
Total conflict resolution cost				2.06

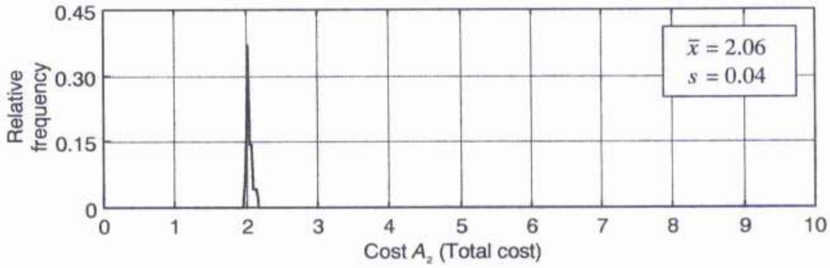
Table 4.2: Values of parameters and costs of the resolution trajectories in Figure 4.10(a)

their statistics are shown in Figure 4.10. It can be noticed that the standard deviations of the samples are significantly smaller than those obtained in the previous chapter when simulating the application of the behaviouristic mechanism in this scenario. This can be justified by the fact that, for the reflective mechanism, the PDFs $f(c_1)$, $f(c_2)$, $f(s_c)$ and $f(d_{A1-A2})$ only encode the random elements inherent to the M-MRMH algorithm. These random elements are expected to result in small standard deviations of the samples obtained, since the plans produced by the algorithm in a given scenario are all approximations of the same plan: the one that results in the minimum value of $w(x)$.

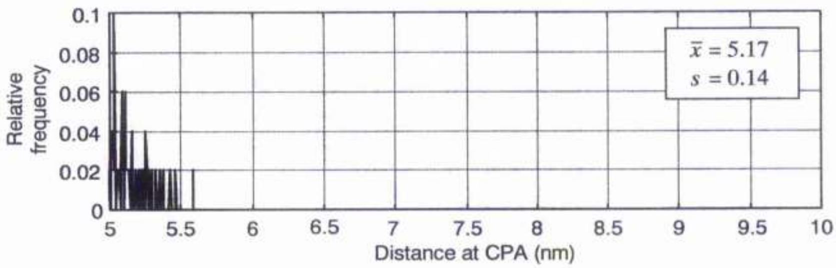
Given the size of the samples obtained, the sample mean can be considered as a point estimate of the mean of the PDF according to which the sample has been drawn. Interval estimates for the mean of that PDF can be formulated based on the values of the sample mean $\bar{x}$ and the sample standard deviation s by applying the Central Limit Theorem. For example, according to the statistics of the sample in Figure 4.11(a), a point estimate of the mean of the PDF $f(c_2)$, which in this case is equal to the PDF $f(s_c)$, is given by the sample mean $\bar{x}$ =2.06. A 95% confidence interval for that mean would be expressed as follows:

$$\left[\bar{x}-1.96\sigma_{\bar{x}}, \bar{x}+1.96\sigma_{\bar{x}}\right] \cong \left[\bar{x}-1.96s, \bar{x}+1.96s\right] = \left[1.98, 2.14\right] \tag{4.22}$$

Considering the sample means in Figure 4.11 and the results obtained for this conflict scenario with the behaviouristic co-operation mechanism, a comparison between point estimates of the means of the four PDFs for the two co-operation mechanisms are shown in Table 4.3.



(a) Relative frequency graph and sample statistics for the cost of A_2's resolution trajectory (total cost)



(d) Relative frequency graph and sample statistics for the predicted distance between the aircraft at the CPA

Figure 4.11. Probabilistic nature of the planning algorithm: repeated running of the M-MRMH algorithm in conflict scenario 1. The algorithm has been run 50 times

	Point estimates of the mean (Conflict scenario 1)	
	Behaviouristic co-operation	**Reflective co-operation**
$f(c_1)$	0.94	0.00
$f(c_2)$	2.88	2.06
$f(s_c)$	3.83	2.06
$f(d_{A1-A2})$	5.84	5.17

Table 4.3: Comparison between point estimates of the means of the PDFs for the two co-operation mechanisms in conflict scenario 1

According to Table 4.3, the reflective co-operation mechanism causes A_2 to bear the entire resolution cost while the behaviouristic co-operation mechanism enables A_1 to share part of that cost. However, the cost borne by A_2 with the behaviouristic mechanism is greater than with the reflective one, and so is the total cost. Thus, in conflict scenario 1 the reflective co-operation mechanism can be said to result, on average, in lower costs for both aircraft than the behaviouristic co-operation mechanism.

4.5.5.3 Application of the reflective co-operation mechanism in conflict scenario 2

This section shows the results of simulating the application of the reflective co-operation mechanism in conflict scenario 2, which is reproduced in Figure 4.12. The resolution trajectories that constitute the joint plan obtained from a run of the algorithm are shown in Figure 4.13 together with the predicted distances between the two aircraft as they fly those resolution trajectories. Table 4.4 displays the parameters defining the two resolution trajectories as well as the cost associated to each of them and their sum. In this case, the joint plan obtained distributes the total resolution cost equitably between the two aircraft, since, as shown in Table 4.4, the resolution trajectories result in both aircraft bearing almost the same cost.

The M-MRMH algorithm has been run 50 times to obtain samples of the random variables c_1, c_2, s_c and $d_{A1\text{-}A2}$ for this scenario. The relative frequency graphs of the samples obtained as well as their statistics are shown in Figure 4.13. As expected, the standard deviations of the samples are significantly smaller than those obtained in the previous chapter when simulating the application of the behaviouristic mechanism in this scenario.

The sample means of the samples obtained can be considered as point estimates of the means of the PDFs according to which the samples have been drawn. Additionally, interval estimates for the means of those PDFs can be formulated based on the values of the sample means and the sample standard deviations by applying the Central Limit Theorem. For example, a point estimate of the mean of the PDF $f(c_1)$ can be given by the sample mean of the sample in Figure 4.14(a), $\bar{x}=2.92$. A 95% confidence interval for that mean would be expressed as follows:

$$\left[\bar{x}-1.96\sigma_{\bar{x}}, \bar{x}+1.96\sigma_{\bar{x}}\right] \cong \left[\bar{x}-1.96s, \bar{x}+1.96s\right] = \left[2.65, 3.19\right] \tag{4.23}$$

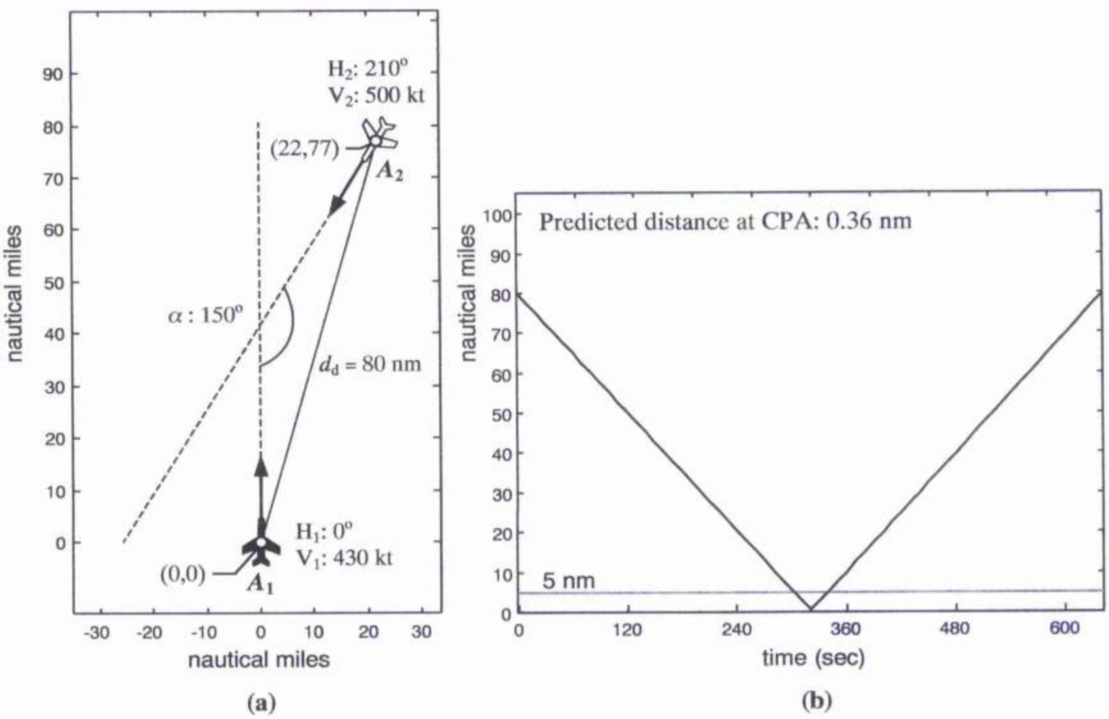


Figure 4.12. Conflict scenario 2. (a) Initial configuration (b) Predicted distances between the aircraft as they fly along their initially intended routes

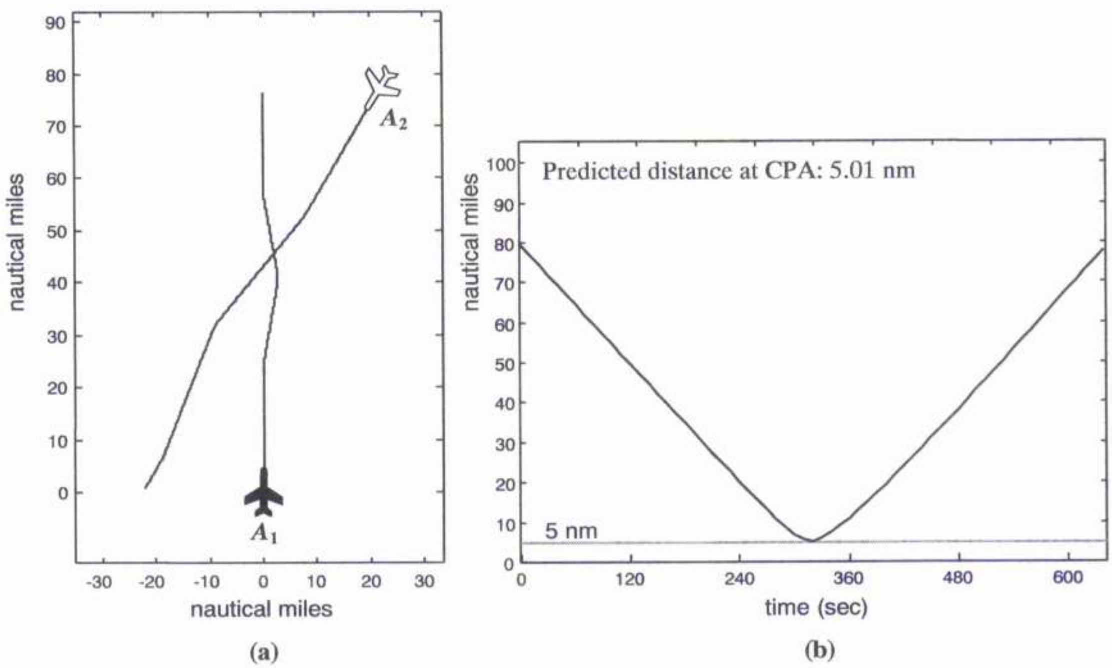


Figure 4.13. Simulation of the application of the M-MRMH planning algorithm in conflict scenario 2. The resulting resolution trajectories are independent of which of the two aircraft is the team organiser. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their assigned resolution trajectories

	γ	t_s	t_e	**Cost**
A_1	10°	204 s	480 s	3.00
A_2	9°	208 s	591 s	2.90
Total conflict resolution cost				5.90

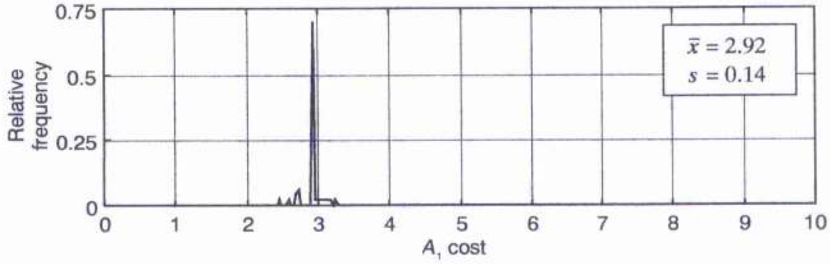
Table 4.4: Values of parameters and costs of the resolution trajectories in Figure 4.13(a)

Considering the sample means in Figure 4.14 and the results obtained for this conflict scenario with the behaviouristic co-operation mechanism, Table 4.5 below shows a comparison between point estimates of the means of the four PDFs for each of the two co-operation mechanisms.

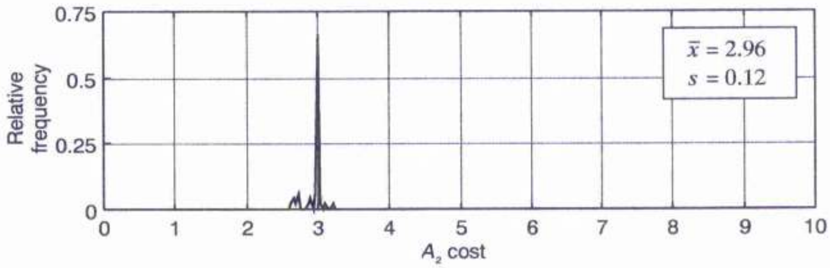
	Point estimates of the mean (Conflict scenario 1)	
	Behaviouristic co-operation	**Reflective co-operation**
$f(c_1)$	2.58	2.92
$f(c_2)$	2.69	2.96
$f(s_c)$	5.27	5.89
$f(d_{A1\text{-}A2})$	5.22	5.26

Table 4.5: Comparison between point estimates of the means of the PDFs for the two co-operation mechanisms in conflict scenario 2

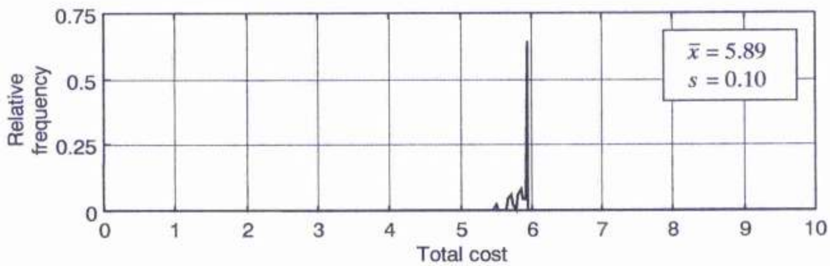
According to Table 4.5, in conflict scenario 2 both co-operation mechanisms enable the two aircraft to share the total resolution cost in an equitable manner. On average, the reflective co-operation mechanism results in a higher total cost than the behaviouristic one but allows for a more even distribution of that cost between the two aircraft. It is worth noting, however, that the total resolution cost resulting from the application of the behaviouristic mechanism in this scenario is expected to be significantly higher than its estimate in Table 4.5 with long response latencies of the conflicting leader's flight crew. This variation in performance, which was discussed in the previous chapter, is caused by the short time lag between conflict detection and CPA in this scenario.



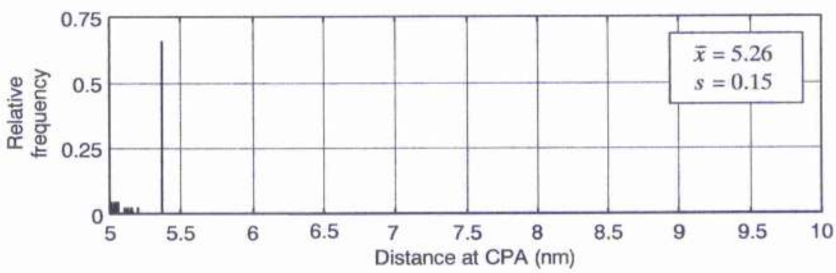
(a) Relative frequency graph and sample statistics for the cost of A_1's resolution trajectory



(b) Relative frequency graph and sample statistics for the cost of A_2's resolution trajectory



(c) Relative frequency graph and sample statistics for the total cost of the conflict resolution (cost of A1's resolution trajectory + cost of A2's resolution trajectory)



(d) Relative frequency graph and sample statistics for the predicted distance between the aircraft at the CPA

Figure 4.14. Probabilistic nature of the planning algorithm: repeated running of the M-MRMH algorithm in conflict scenario 2. The algorithm has been run 50 times

4.5.5.4 Comparison between the performance of the reflective co-operation mechanism in conflict scenarios 1 and 2

The differences in the performance of the reflective co-operation mechanism in the two conflict scenarios considered can be partly explained by the fact that the time lag between conflict detection and CPA is much shorter in conflict scenario 2 than in conflict scenario 1. It is expected that the shorter the time lag between conflict detection and CPA, the more the aircraft will have to deviate from their initially intended routes to resolve their conflict, as larger turns will be required to achieve safe separation. The effect of the time between conflict detection and CPA on the resolution trajectories is accentuated by the fact that the joint plans do not schedule the aircraft to start manoeuvring until at least 200 seconds after the detection of the conflict. In conflict scenario 1, the time lag between conflict detection and CPA is approximately 1140 seconds. This time lag allows for joint plans that resolve the conflict with a low cost resolution trajectory flown by A_2. On the other hand, the time lag between conflict detection and CPA in conflict scenario 2 is approximately 320 seconds. In this case, the joint plans require both aircraft to manoeuvre to resolve the conflict.

4.5.5.5 Comparative analysis of the performance of the reflective and behaviouristic co-operation mechanisms in various conflicting configurations

In this section, the performance of the reflective co-operation mechanism in different conflict scenarios will be investigated and compared with the performance of the behaviouristic co-operation mechanism in those same scenarios. The conflict scenarios considered are those in which the performance of the behaviouristic mechanism was analysed in the section 3.4.5.10 of the previous chapter. As it was explained in that section, the scenarios studied are defined by assigning different values to the parameters V_1, V_2, d_d, and α in Figure 4.15.

The resolution costs resulting from the application of the reflective co-operation mechanism in several conflict scenarios of the kind shown in Figure 4.15 will be analysed statistically. The M-MRMH algorithm has been repeatedly run 50 times for each of the 18 scenarios that result from successively assigning to α the values

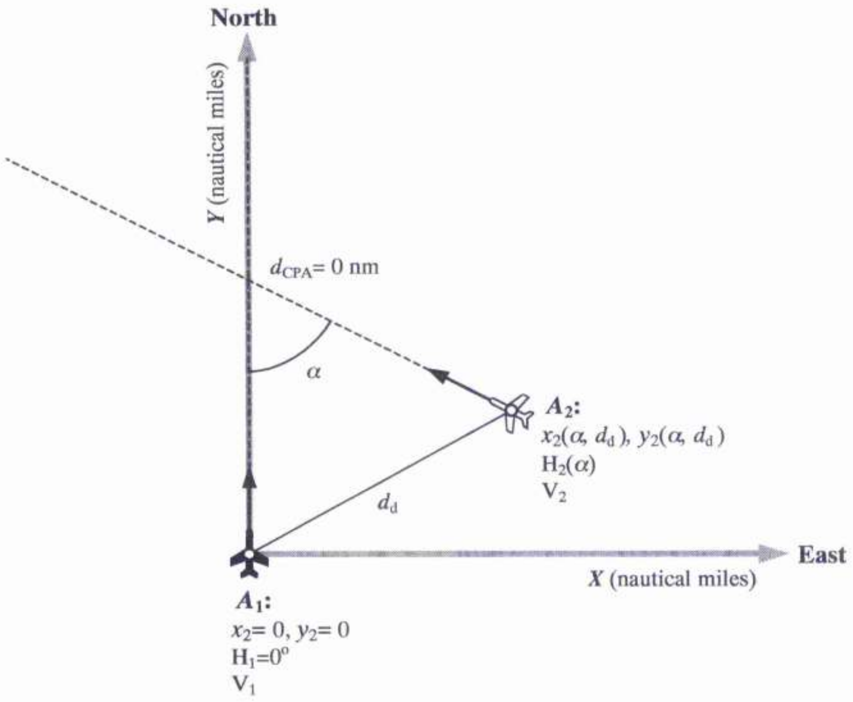


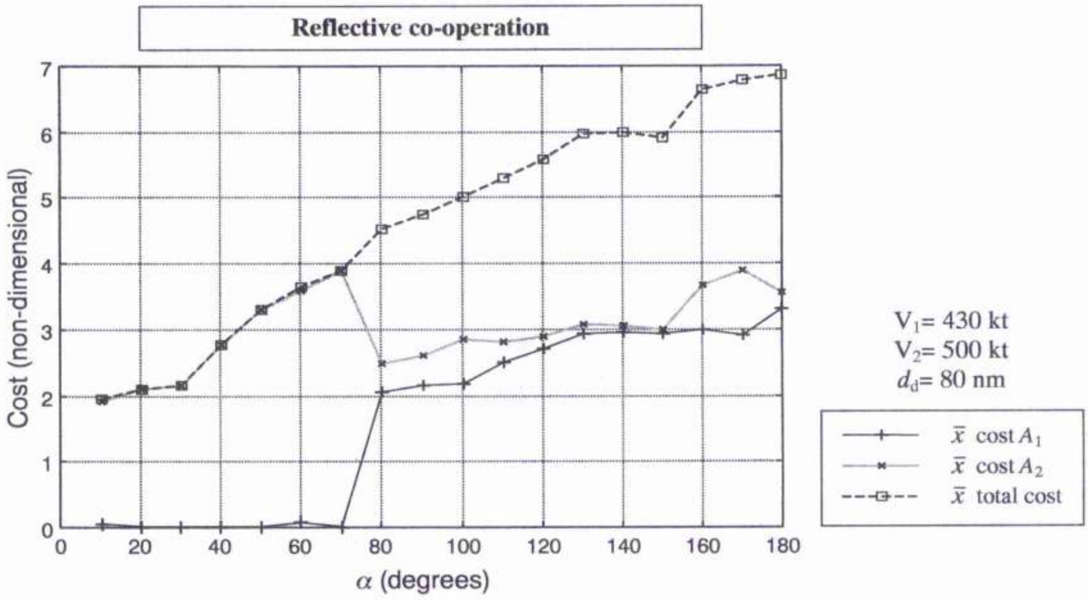
Figure 4.15. Conflict scenario defined from a given set of values of the parameters V_1, V_2, α and d_d

10°, 20°, 30°, … , 160°, 170°, 180°, while assuming fixed values of V_1, V_2 and d_d. Each series of runs results in a sample from each of the four random variables c_1, c_2, s_c and $d_{A1\text{-}A2}$ for the corresponding conflict scenario. Only the samples from c_1, c_2, and s_c are considered in this analysis. These samples are drawn according to the PDFs $f(c_1)$, $f(c_2)$ and $f(s_c)$, respectively.

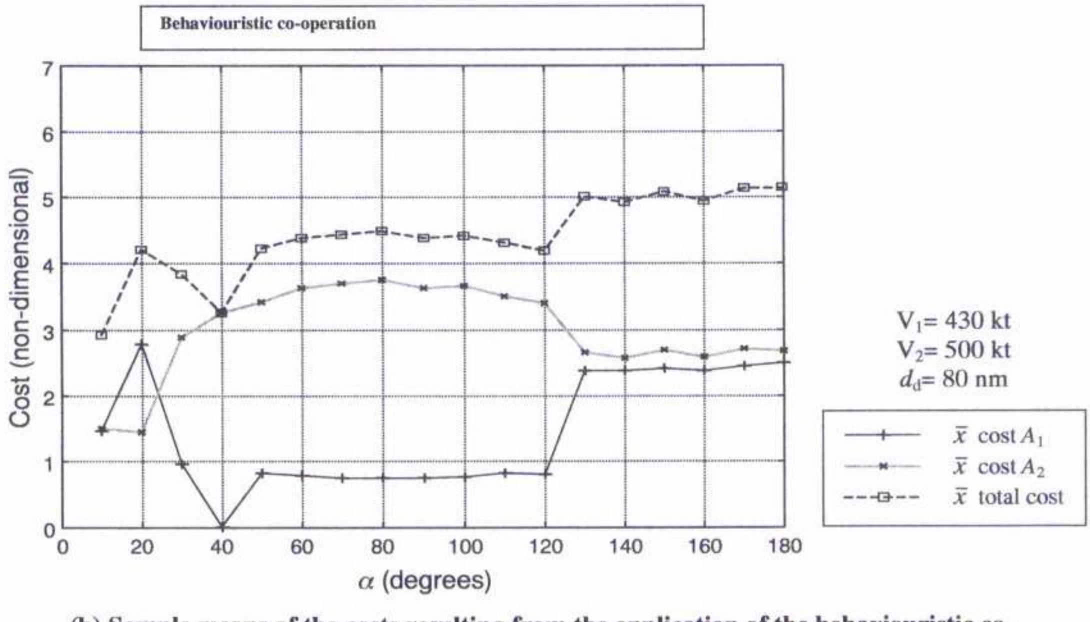
To illustrate the performance of the reflective co-operation mechanism for the different values of α, the means of the samples obtained, which are considered as point estimates of the means of the corresponding PDFs, are plotted against α. One of such plots has been generated for each of the four combinations of values for the parameters V_1, V_2 and d_d considered in the section 3.4.5 of the previous chapter. These combinations are: V_1= 430 kt, V_2= 500 kt and d_d= 80 nm; V_1= 430 kt, V_2= 500 kt and d_d= 90 nm; V_1= 465 kt, V_2= 500 kt and d_d= 80 nm; V_1= 465 kt, V_2= 500 kt and d_d= 90 nm. With the objective of comparing the performance of the two co-operation mechanisms, estimates of the means of the PDFs $f(c_1)$, $f(c_2)$ and $f(s_c)$ obtained from repeated simulations of the application of the behaviouristic co-operation mechanism in the scenarios considered have also been plotted against α. In these simulations, no assumption is made about which of the two aircraft acts as the leader and, consequently, the sequence of aircraft actions is random.

Figure 4.16 shows the two plots obtained for V_1 = 430 kt, V_2 = 500 kt and d_d = 80 nm. According to Figure 4.16(a), which corresponds to the reflective co-operation mechanism, for $\alpha < 80^\circ$ the joint plan does not generally require A_1 to modify its initially intended route. On the other hand, for values of $\alpha \geq 80^\circ$, both aircraft modify their intended trajectories to resolve the conflict. In these cases, the total resolution cost is distributed evenly between the two aircraft. It can also be noted from Figure 4.16(a) that the total resolution cost grows steadily as α increases. The variations in the performance of the M-MRMH algorithm for the different values of α can be explained by the fact that, as α increases, the resulting conflict scenarios present shorter times between conflict detection and CPA and, consequently, larger manoeuvres may be required to achieve safe separation. For small values of α, A_2 generally resolves the conflict by itself without incurring excessive cost, while for large values of α the two aircraft are required to modified their initially intended route.

Comparing Figure 4.16(a) with Figure 4.16(b) it can be seen that while the behaviouristic mechanism allows for an even distribution of the total cost between the two aircraft for $\alpha \geq 130^\circ$, the reflective one does so for $\alpha \geq 80^\circ$. For $\alpha < 80^\circ$ the behaviouristic mechanism generally enables A_1 to share part of the resolution cost, while the reflective one generally results in A_2 bearing the entire resolution cost. While the total resolution cost steadily grows with α for the reflective co-operation mechanism, it does not vary greatly with α for the behaviouristic one, except for a small increase with $\alpha \geq 130^\circ$. As a result, the total resolution cost with the behaviouristic mechanism is higher than with the reflective one for small values of α and lower than with the reflective one for large values of α. This indicates that the effect of the time between conflict detection and CPA influences more the performance of the reflective co-operation mechanism than it does that of the behaviouristic one. This different degree of influence can be justified by the fact that, while the behaviouristic mechanism allows the aircraft to start manoeuvring as soon as 120 seconds after they detect the conflict, the reflective co-operation mechanism does not do so until at least 200 seconds after.



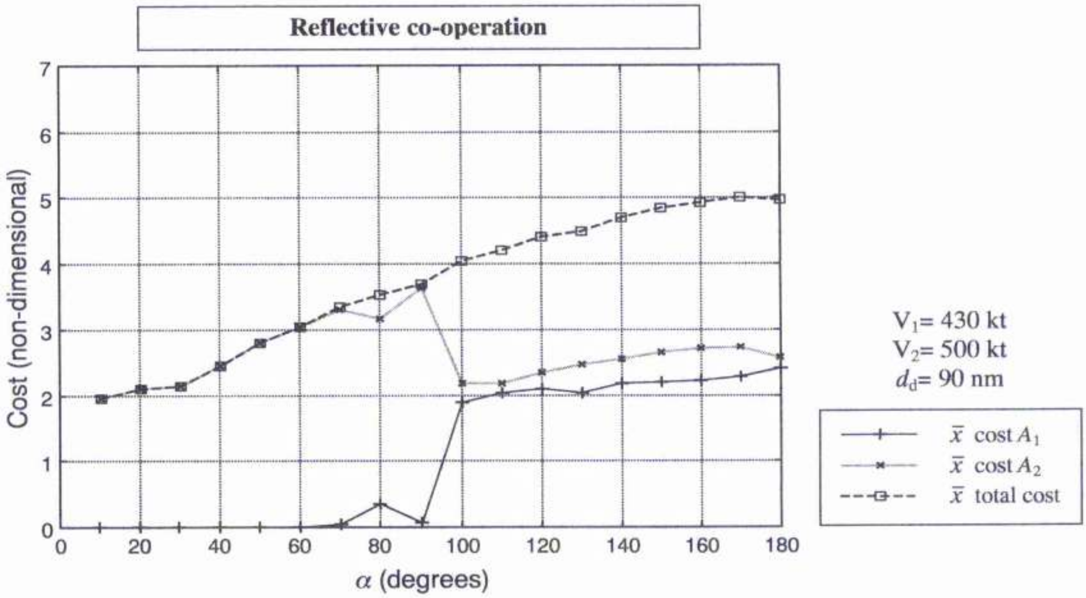
(a) Sample means of the costs resulting from the application of the reflective co-operation mechanism



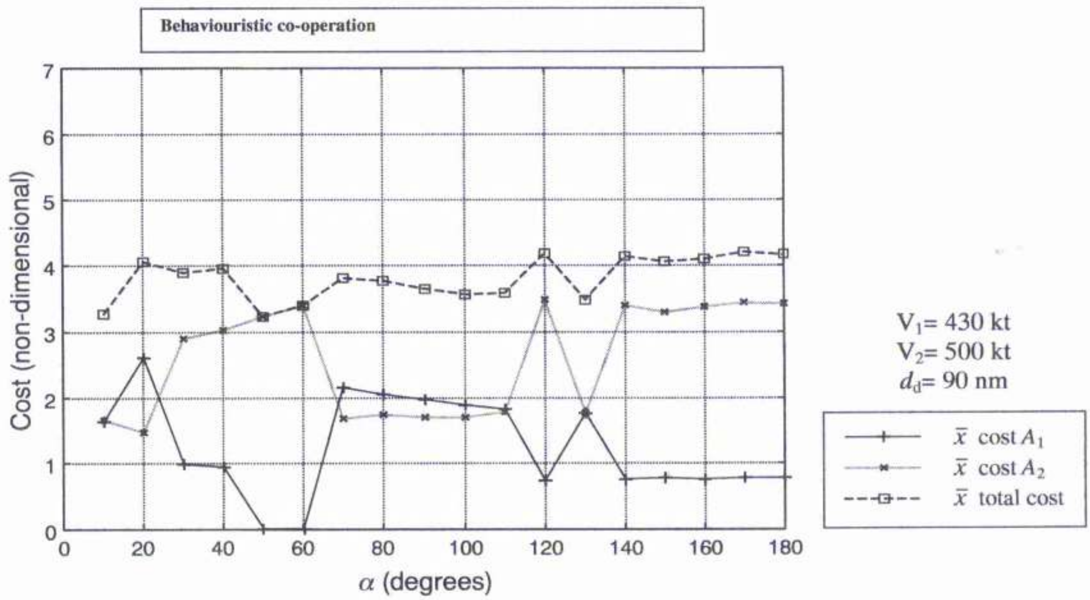
(b) Sample means of the costs resulting from the application of the behaviouristic co-operation mechanism without specifying which aircraft acts as the leader

Figure 4.16. Comparison of the performance between the behaviouristic and the reflective co-operation mechanisms in two-aircraft conflict scenarios for different values of the path crossing angle α given d_d= 80 nm, V_1 =430 kt and V_2= 500 kt. The size of the random samples drawn is 50

The plots in Figure 4.17 are those obtained for V_1 = 430 kt, V_2 = 500 kt and d_d = 90 nm. In this case, the aircraft's speeds are the same as in Figure 4.16 but the conflicts are detected earlier as a larger ADS-B range of coverage is assumed. Comparing Figures



(a) Sample means of the costs resulting from the application of the reflective co-operation mechanism



(b) Sample means of the costs resulting from the application of the behaviouristic co-operation mechanism without specifying which aircraft acts as the leader

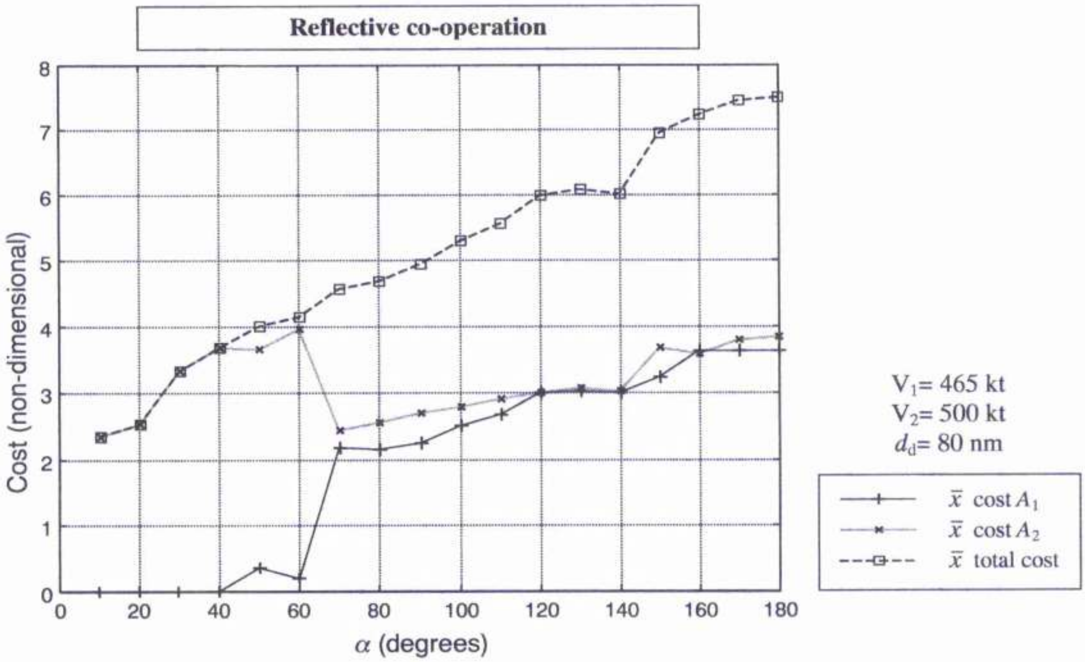
Figure 4.17. Comparison of the performance between the behaviouristic and the reflective co-operation mechanisms in two-aircraft conflict scenarios for different values of the path crossing angle α given d_d= 90 nm, V_1 =430 kt and V_2= 500 kt. The size of the random samples drawn is 50.

4.17(a) and 4.16(a), it can be seen how the earlier conflict detection influences the performance of the mechanism for the speeds considered. The total resolution cost grows as α increases in both cases, but is generally lower with d_d = 90 nm, especially for large values of α. This is explained by the fact that, for a given conflict, a larger value of d_d implies a longer time lag between conflict detection and the CPA, which allows the aircraft to start manoeuvring before and resolve the conflict without deviating excessively from their initially intended routes. As a consequence, the conflicts can be resolved without A_1 having to manoeuvre for larger values of α with d_d = 90 nm than with d_d = 80 nm.

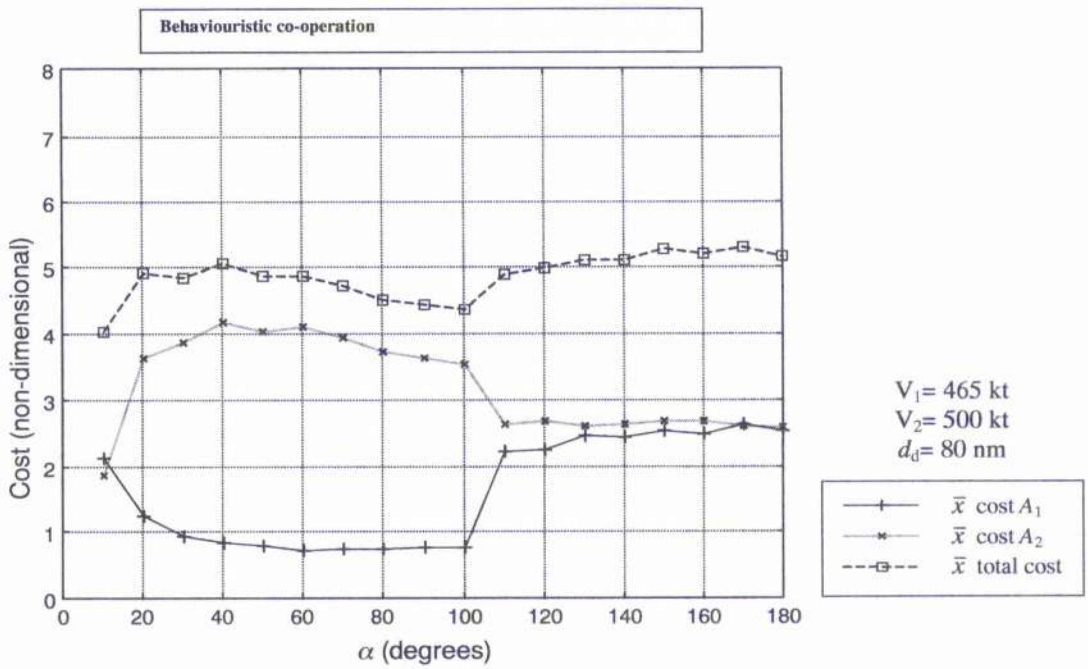
Comparing Figures 4.17(a) and 4.17(b), it can be seen that, while the reflective co-operation mechanism enables the two aircraft to share in an equitable manner the total resolution cost for $\alpha \geq 100°$, the behaviouristic one only does so for some values of α, mainly between 70° and 130°. The behaviouristic co-operation mechanism, whose performance is seemingly unaffected by the variations of α in this case, again results in a higher total cost than the reflective one for small values of α and a lower total cost than the reflective one for large values of α.

Figure 4.18 shows the two plots obtained for V_1 = 465 kt, V_2 = 500 kt and d_d = 80 nm. In this case, the values of V_2 and d_d are the same as in Figure 4.16 and V_1 is increased from 430 kt to 465 kt. Comparing Figure 4.18(a) with Figure 4.16(a), it can be seen that the variations in the performance of the reflective co-operation mechanism with α display similar trends in both cases. However, the total resolution cost is generally higher with V_1 = 465 kt than with V_1 = 430 kt. While with V_1 = 430 kt the reflective co-operation mechanism distributes evenly the total resolution cost between the two aircraft for $\alpha \geq 80°$, with V_1 = 465 kt it does so for $\alpha \geq 70°$. Thus, the conflicts can be resolved without A_1 having to manoeuvre for larger values of α with V_1 = 430 kt than with V_1 = 465 kt.

Comparing Figures 4.18(a) and 4.18(b), it can be seen that the reflective co-operation mechanism enables the two aircraft to share the total resolution cost equitably for $\alpha \geq 70°$, while the behaviouristic one only does so for $\alpha = 10°$ and $\alpha \geq 110°$. However, the behaviouristic co-operation mechanism enables A_1 to share part of the total cost for all the values of α considered, unlike the reflective one. The performance of the behaviouristic mechanism does not vary greatly with α. This results in higher total costs than the reflective one for small values of α and lower total costs than the reflective one for large values of α.



(a) Sample means of the costs resulting from the application of the reflective co-operation mechanism



(b) Sample means of the costs resulting from the application of the behaviouristic co-operation mechanism without specifying which aircraft acts as the leader

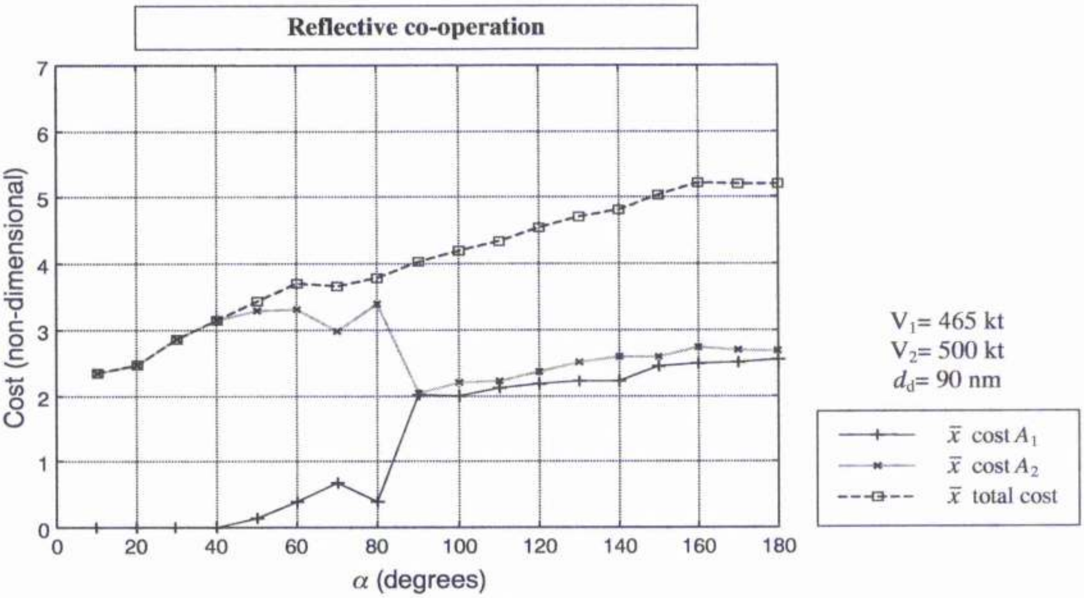
Figure 4.18. Comparison of the performance between the behaviouristic and the reflective co-operation mechanisms in two-aircraft conflict scenarios for different values of the path crossing angle α given d_d= 80 nm, V_1 =465 kt and V_2= 500 kt. The size of the random samples drawn is 50

The plots in Figure 4.19 are those obtained for V_1 = 465 kt, V_2 = 500 kt and d_d = 90 nm. The total cost of the application of the reflective co-operation mechanism grows steadily with α, as it can be seen in Figure 4.19(a). The total costs in Figure 4.19(a) are lower than those in Figure 4.18(a) for almost all values of α. This can be explained by the greater value of d_d in the conflict scenarios considered in Figure 4.19(a). The reflective mechanism results in both aircraft having to manoeuvre for $\alpha \geq 90^o$ instead of for $\alpha \geq 70^o$ which is the case in Figure 4.18(a). This is also due to the greater value of d_d in Figure Figure 4.19(a). The total costs in Figure 4.19(a) are, in general, slightly higher than those in Figure 4.17(a). This was expected as a result of the larger value of V_1 in Figure 4.19(a). For this same reason, in Figure 4.17(a) the total resolution cost is evenly distributed between the two aircraft only for $\alpha \geq 100^o$.

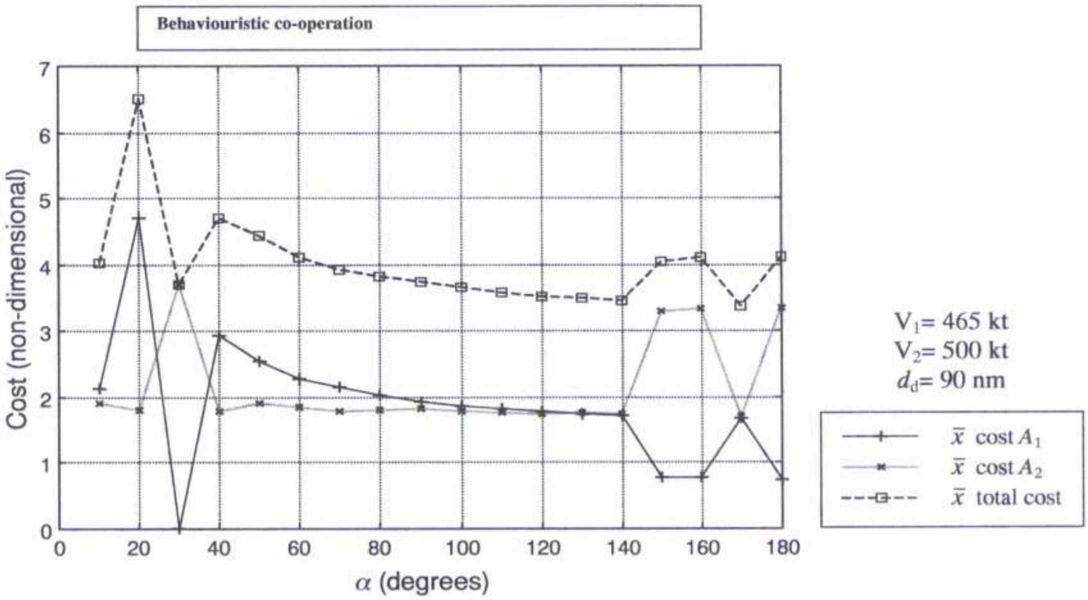
Comparing Figures 4.19(a) and 4.19(b), it can be seen that, while the reflective co-operation mechanism enables the two aircraft to share the total resolution cost equitably manner for $\alpha \geq 90^o$, the behaviouristic one does so for most values of α. Again, the total resolution costs obtained with the behaviouristic co-operation mechanism are higher than those obtained with the reflective one for small values of α and lower for large values of α.

Comparing Figures 4.16(a), 4.17(a), 4.18(a) and 4.19(a), it can be concluded that with d_d = 90 nm the reflective co-operation mechanism generally results in lower total resolution costs than with d_d = 80 nm. However, with d_d = 90 nm there are less scenarios in which the mechanism results in an equitable distribution of the total cost between the two aircraft. Additionally, it can also be said that with V_1 = 430 kt and V_2 = 500 kt the reflective co-operation mechanism generally results in lower total resolution costs than with V_1 = 465 kt and V_2 = 500 kt. However, with V_1 = 430 kt and V_2 = 500 kt there are less scenarios in which the mechanism results in an equitable distribution of the total cost between the two aircraft.

Comparing Figures 4.16(b), 4.17(b), 4.18(b) and 4.19(b), it can be concluded that with d_d = 90 nm the behaviouristic co-operation mechanism generally results in lower total resolution costs than with d_d = 80 nm, as is the case for the reflective one. It can also be



(a) Sample means of the costs resulting from the application of the reflective co-operation mechanism



(b) Sample means of the costs resulting from the application of the behaviouristic co-operation mechanism without specifying which aircraft acts as the leader

Figure 4.19. Comparison of the performance between the behaviouristic and the reflective co-operation mechanisms in two-aircraft conflict scenarios for different values of the path crossing angle α given d_d= 90 nm, V_1 =465 kt and V_2= 500 kt. The size of the random samples drawn is 50

said that with V_1 = 430 kt and V_2 = 500 kt there are generally less conflict scenarios in which the behaviouristic co-operation mechanism results in an equitable distribution of the total cost between the two aircraft than with V_1 = 465 kt and V_2 = 500 kt, which was also the case for the reflective one. Unlike the reflective co-operation mechanism, the number of scenarios in which the behaviouristic one results in an equitable distribution of the total cost between the two aircraft cannot be said to be generally smaller with d_d = 90 nm than with d_d = 80 nm.

Applying symmetry considerations, the results obtained in this section can be extended to values of α between 180^o and 360^o.

4.5.6 Two-dimensional conflicts involving three aircraft

This section illustrates the application of the reflective co-operation mechanism with the M-MRMH planning algorithm in two-dimensional conflict scenarios involving three aircraft. With the objective of comparing the performance of the two co-operation mechanisms, the three-aircraft conflict scenarios investigated in this section are the same as those for which the performance of the behaviouristic mechanism was analysed in the previous chapter. To comply with the assumptions introduced in section 4.4.4, the initial positions of the aircraft in some of those conflict scenarios will have to be modified to make them correspond to the time when the first team membership proposal including the three aircraft is issued. This time coincides with the time from which one of the three aircraft is simultaneously within the ADS-B range of coverage of the other two.

4.5.6.1 Simulation of the application of the reflective co-operation mechanism

The conflict resolution process starts when one of the aircraft is able to detect all the conflicts in the scenario and sends a team membership proposal message to the other two conflicting aircraft. Once the team has been established, the team organiser runs the M-MRMH algorithm to elaborate a joint plan and, subsequently, it presents its flight crew with its assigned resolution trajectory and transmits to the other team members their respective resolution trajectories. The three flight crews accept their respective resolution trajectories within 200 seconds of the time when the team organiser issues the team membership proposal. The resolution trajectories in the plans produced by the M-MRMH are independent of the response times of the flight crews, as they do not require the aircraft to alter their intended routes within 200 seconds of the initial time.

Additionally, they are also independent of whether the team organiser's team mates are initially within each other's ADS-B range of coverage or not. As a consequence of this, the distinction between conflict scenarios of types I and II is irrelevant in this case.

The M-MRMH algorithm has been adapted specifically to two-dimensional conflict scenarios involving three aircraft considering the assumptions discussed in section 4.4.4. The function $w(\boldsymbol{x})$ to be minimised by the algorithm in such conflicts is of the form:

$$w(\boldsymbol{x}) = \frac{1}{3}C_1(\gamma^1, t_s^1, t_e^1) + \frac{1}{3}C_2(\gamma^2, t_s^2, t_e^2) + \frac{1}{3}C_3(\gamma^3, t_s^3, t_e^3) \tag{4.24}$$

where the three weights w_1, w_2 and w_3 have been set to $\frac{1}{3}$. These values of the weights comply with the requirements w_1, w_2, $w_3 > 0$ and $w_1 + w_2 + w_3 = 1$.

The minimum $\boldsymbol{x}^*$ of $w(\boldsymbol{x})$ is also the minimum of the function $C_t(\boldsymbol{x}) = 3w(\boldsymbol{x}) = C_1(\gamma^1, t_s^1, t_e^1) + C_2(\gamma^2, t_s^2, t_e^2) + C_3(\gamma^3, t_s^3, t_e^3)$, which is the total cost of resolving the conflict. Hence, besides belonging to the Pareto set of the multi-objective minimisation problem involving the three individual cost functions, the plan $\boldsymbol{x}^*$ results in the lowest possible total resolution cost.

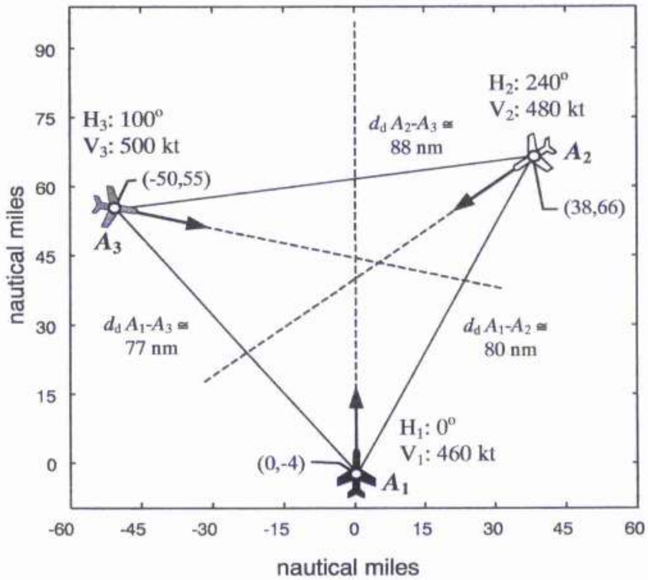
As it was done for two-dimensional conflicts involving two aircraft, the MATLAB® integrated computing environment has been used to implement and run the algorithm, as well as to visualise the resulting resolution trajectories. The version of MATLAB® and the computing platform used are again those described in sections 4.4.5 and 3.4.5. The CPU time limit for the iterative improvement processes performed by the M-MRMH algorithm, t_{it}, has been set to 30 seconds for three-aircraft conflicts, three times longer than for two-aircraft conflicts. This is justified by the fact that, in order to check whether a candidate plan is conflict-free in a three-aircraft conflict, the algorithm has to perform three times as many computations as in a two-aircraft one. Again, it is assumed that the value of the time limit t_{it} in an hypothetical airborne implementation of the M-

MRMH algorithm could be reduced to ensure that a joint plan is produced and sanctioned by all the flight crews within the established 200 seconds limit.

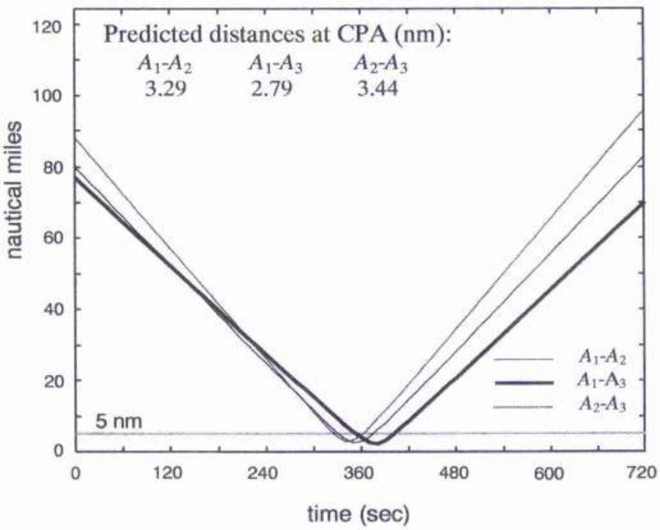
The performance of the reflective co-operation mechanism in the conflict scenarios considered will be analysed statistically. To do so, the M-MRMH algorithm will be repeatedly run for each of those scenarios to obtain samples of the random variables $\boldsymbol{c_1}$, $\boldsymbol{c_2}$, $\boldsymbol{c_3}$, $\boldsymbol{s_c}$, $\boldsymbol{d_{A1\text{-}A2}}$, $\boldsymbol{d_{A1\text{-}A3}}$ and $\boldsymbol{d_{A2\text{-}A3}}$. These random variables, which are defined for each of the scenarios considered, describe, respectively, the resolution cost incurred by each of the three aircraft, the total resolution cost and the minimum predicted distances between the aircraft as they fly along their resolution trajectories. The samples obtained from the repeated runs of the algorithm are drawn according to the PDFs $f(c_1)$, $f(c_2)$, $f(c_3)$, $f(s_c)$, $f(d_{A1\text{-}A2})$, $f(d_{A1\text{-}A3})$ and $f(d_{A2\text{-}A3})$, respectively. These PDFs are not influenced by the response latencies of the flight crews and reflect only the probabilistic nature of the M-MRMH algorithm.

4.5.6.2 Application of the reflective co-operation mechanism in conflict scenarios 3 and 4

The ADS-B range of coverage in conflict scenario 3 is assumed to be 80 nm. The initial positions of the aircraft in this scenario when applying the behaviouristic co-operation mechanism, which were shown in Figure 3.33(a), correspond to the time when A_2 and A_3 enter each other's area of ADS-B coverage. From that time on, the three aircraft are within the ADS-B range of coverage of one another. When applying the reflective co-operation mechanism the initial positions of the aircraft correspond to the time when A_1 and A_2 enter each other's area of ADS-B coverage. From that time on, one of the three aircraft, A_1 in this case, is within the ADS-B range of coverage of the other two and, consequently, is able to detect all the conflicts in the scenario. The positions of the aircraft at that time, which are shown in Figure 4.20(a), are their positions 33 seconds before they reach the configuration displayed in Figure 3.33(a). The co-ordinates of the aircraft in Figure 4.20(a) have been rounded to the nearest nautical mile for the sake of clarity.



(a)



(b)

Figure 4.20. Conflict scenario 3. (a) Initial configuration (b) Predicted distances between the aircraft as they fly along their initially intended routes

A_1 is the aircraft that first becomes aware of the presence of the other two in conflict scenario 3. As soon as it does so, A_1 sends a team membership proposal message to both A_2 and A_3, which are not within each other's ADS-B range of coverage yet. Since its team membership proposal includes the three aircraft in the scenario, A_1 acts as the team organiser. The three flight crews eventually accept to join the team and A_1 applies the M-MRMH algorithm to elaborate a joint conflict resolution plan. The resolution trajectories that form the joint plan obtained from a run of the algorithm are shown in Figure 4.21, together with the predicted distances between the aircraft as they fly those resolution trajectories. Figure 4.22 shows a sequence of snapshots of the positions of the aircraft along the resolution trajectories. Table 4.6 displays the parameters defining the three resolution trajectories as well as the cost associated to each of them and their sum, which is regarded as the total cost of resolving the conflict.

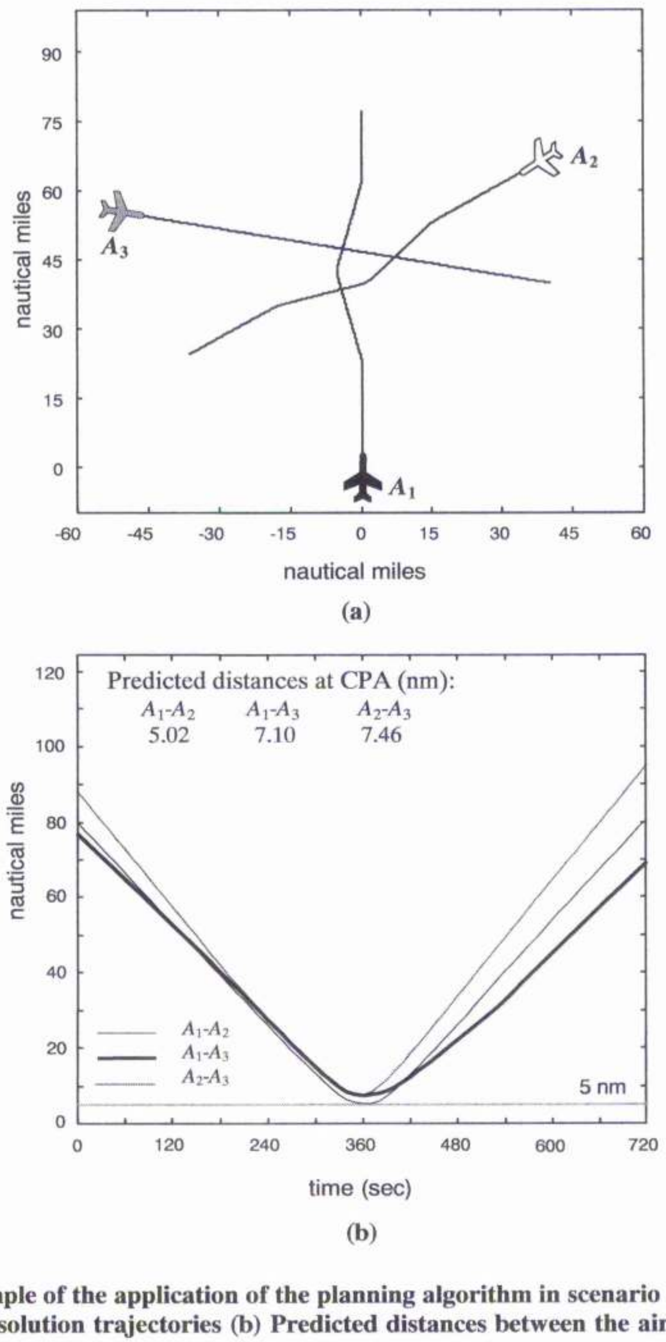


Figure 4.21. Example of the application of the planning algorithm in scenario 3. A_1 acts as the team organiser. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

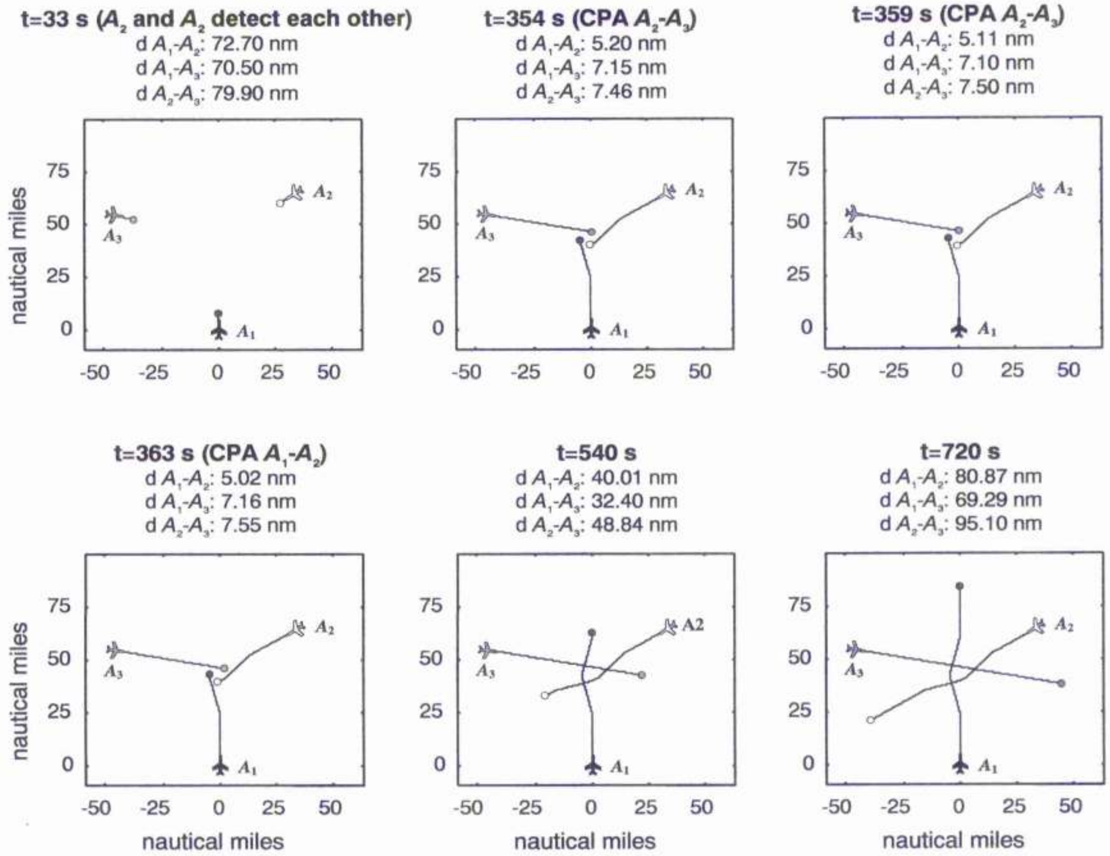


Figure 4.22. Example of the application of the planning algorithm in scenario 3. A_1 acts as the team organiser. Sequence of predicted future positions of the aircraft along the resolution trajectories in Figure 4.21(a)

	γ	t_s	t_e	Cost
A_1	-15°	200 s	530 s	5.19
A_2	-15°	200 s	490 s	5.07
A_3	Initially intended route			0.00
Total conflict resolution cost				10.26

Table 4.6. Example of the application of the planning algorithm in scenario 3. A_1 acts as the team organiser. Values of parameters and costs for the resolution trajectories in Figure 4.21(a)

To obtain samples of the random variables c_1, c_2, c_3, s_c, $d_{A1\text{-}A2}$, $d_{A1\text{-}A3}$ and $d_{A2\text{-}A3}$ for this scenario, the M-MRMH algorithm has been run 50 times. The samples reflect the probabilistic character of the algorithm and are drawn according to the PDFs $f(c_1)$, $f(c_2)$, $f(s_c)$, and $f(d_{A1\text{-}A2})$, respectively. The means of the samples obtained are displayed in

Table 4.7. This table shows that the application of the reflective co-operation mechanism in this scenario results in A_3 maintaining its initially intended route and A_1 and A_2 sharing the total resolution cost.

To identify the differences between the performance of the two co-operation mechanisms in conflict scenario 3, Table 4.7 is compared with Table 3.15, which displays the sample means resulting from the repeated simulation of the application of the behaviouristic mechanism in this conflict scenario. The comparison shows that the total resolution cost is lower with the behaviouristic mechanism for three of the six possible sequences of action. It is also observed that, while the application of the reflective co-operation mechanism results in A_3 not manoeuvring, that is not always the case with the behaviouristic mechanim. In this scenario, the behaviouristic mechanism results in the leader aircraft bearing most of the total resolution cost, while the reflective mechanism enables two of the conflicting aircraft to share the total cost in an equitable manner.

$\bar{x}$ Cost	A_1	5.18
	A_2	5.07
	A_3	0.00
	Total	10.25
$\bar{x}$ CPA distance (nm)	A_1-A_2	5.02
	A_1-A_3	7.10
	A_2-A_3	7.46

Table 4.7. Sample means of the resolution costs and the distances between the aircraft at their CPA for samples obtained simulating 50 times the application of the planning algorithm in conflict scenario 3

The application of the reflective mechanism has been simulated for conflict scenario 4 with the objective of illustrating the influence of the ADS-B range of coverage on the performance of the mechanism. Conflict scenario 4 represents the same conflicting configuration as scenario 3 with an ADS-B range of coverage of 90 nm instead of 80 nm. The initial positions of the aircraft in scenario 4 when applying the behaviouristic co-operation mechanism, which were shown in Figure 3.46(a), correspond to the time when A_2 and A_3 enter each other's area of ADS-B coverage.

When applying the reflective co-operation mechanism, the initial positions of the aircraft correspond to the time when A_1 and A_2 enter each other's area of ADS-B coverage. The positions of the aircraft at that time, which are shown in Figure 4.23(a), are their positions 37 seconds before they reach the configuration displayed in Figure 3.46(a). The co-ordinates of the aircraft in Figure 4.23(a) have been rounded to the nearest nautical mile for the sake of clarity.

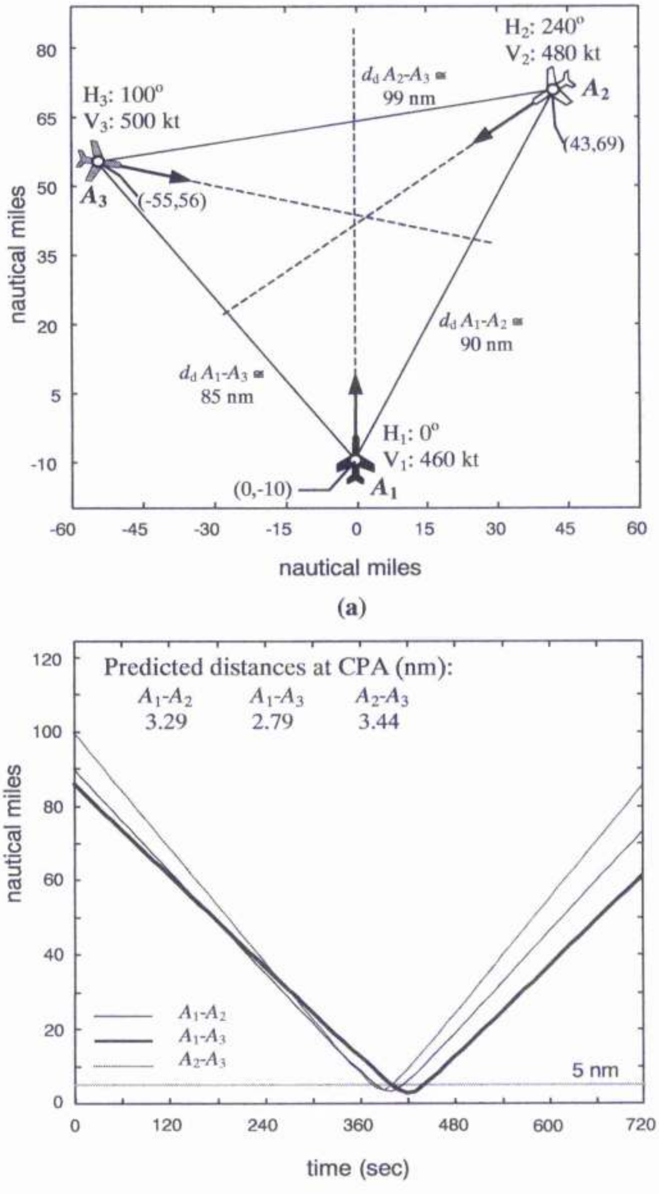


Figure 4.23. Conflict scenario 4. (a) Initial configuration (b) Predicted distances between the aircraft as they fly along their initially intended routes

The team formation process in this conflict scenario is analogous to the one in conflict scenario 3. Thus, A_1 acts as the team organiser in a team formed by the three conflicting aircraft. The M-MRMH algorithm has been repeatedly run 50 times, resulting in samples from the random variables c_1, c_2, c_3, s_c, $d_{A1\text{-}A2}$, $d_{A1\text{-}A3}$ and $d_{A2\text{-}A3}$ for this scenario. The means of the samples obtained are shown in Table 4.8.

$\bar{x}$ Cost	A_1	4.15
	A_2	4.48
	A_3	0.00
	Total	8.63
$\bar{x}$ CPA distance (nm)	A_1-A_2	5.26
	A_1-A_3	6.69
	A_2-A_3	7.96

Table 4.8. Sample means of the resolution costs and the distances between the aircraft at their CPA for samples obtained simulating 50 times the application of the planning algorithm in conflict scenario 4

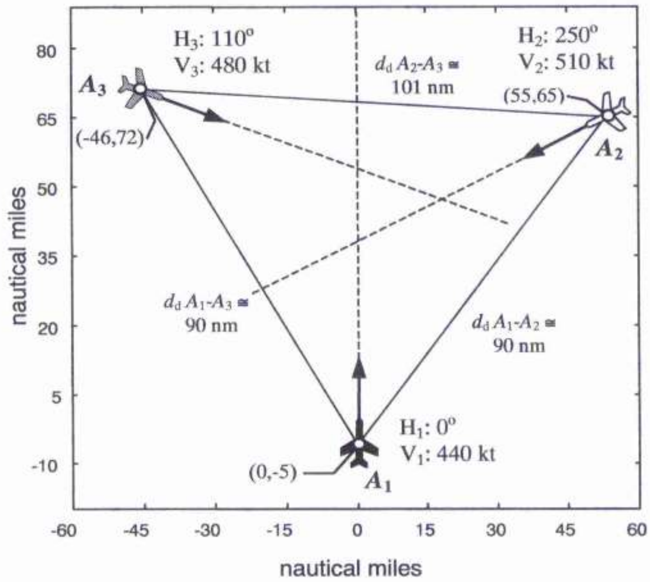
As is the case in conflict scenario 3, the application of the reflective co-operation mechanism in conflict scenario 4 results in A_3 maintaining its initially intended trajectory and A_1 and A_2 sharing equitably the total resolution cost. Comparing Tables 4.8 and 4.7, it can be seen that the resolution costs are lower in conflict scenario 4 than in conflict scenario 3.

To identify the differences between the performance of the two co-operation mechanisms in conflict scenario 4, Table 4.8 is compared with Table 3.16, which displays the sample means resulting from the repeated simulation of the application of the behaviouristic mechanism in this conflict scenario. The comparison leads to similar conclusions to those reached for conflict scenario 3. In this case, the total resolution cost is lower with the behaviouristic mechanism for four of the six possible sequences of action. The application of the reflective co-operation mechanism in conflict scenario 4 results in A_3 not manoeuvring, which is not always the case with the behaviouristic one. The application of the behaviouristic co-operation mechanism in conflict scenario 4 results in one of the aircraft bearing most of the total resolution cost, while the reflective one enables two of them to share the total cost equitably.

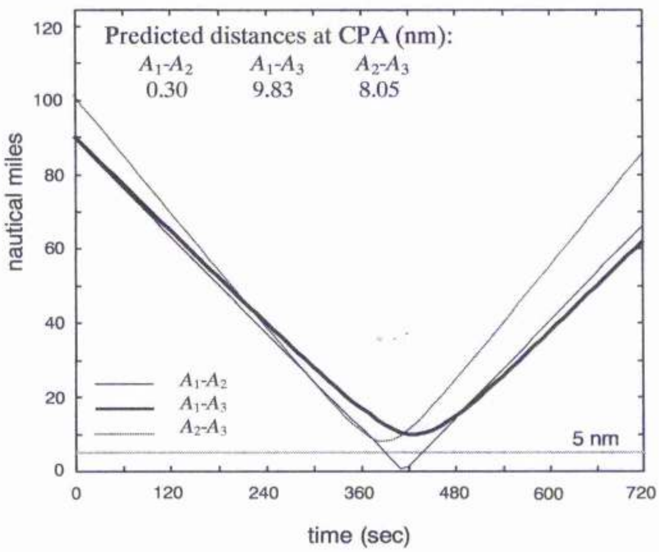
4.5.6.3 Application of the reflective co-operation mechanism conflict scenario 5: two-aircraft conflict in a three-aircraft scenario

This section illustrates how the reflective co-operation mechanism proposed in this chapter can be successfully applied in three-aircraft conflict scenarios in which one of the aircraft is not involved in any conflict. To do so, an example of such a conflict scenario, conflict scenario 5, is considered. The ADS-B range of coverage in this scenario is assumed to be 90 nm. The initial positions of the aircraft in this scenario when applying the behaviouristic co-operation mechanism, which were shown in Figure 3.47(a), correspond to the time when A_2 and A_3 enter each other's area of ADS-B coverage. From that time on, the three aircraft are within the ADS-B range of coverage of one another. When applying the reflective co-operation mechanism, the initial positions of the aircraft correspond to the time when A_1 and A_2 enter each other's area of ADS-B coverage, which in this case is also the time when A_1 and A_3 enter each other's area of ADS-B coverage. From that time on, A_1 is within the ADS-B range of coverage of the other two aircraft. The positions of the aircraft at that time, which are shown in Figure 4.24(a), are their positions 45 seconds before they reach the configuration displayed in Figure 3.47(a). The co-ordinates of the aircraft in Figure 4.24(a) have been rounded to the nearest nautical mile for the sake of clarity.

As soon as A_1 becomes aware of the presence of A_3, it decides to form a team that includes the three aircraft. However, since A_3 is not in conflict with either A_1 or A_2, it cannot be expected to join the team. Thus, A_1 only sends a team membership proposal message to A_2. Nevertheless, A_3 is in the list of A_1's potential team mates included in the message, so that A_2 realises that it does not have complete knowledge of the conflicting situation and A_1 is guaranteed to act as the team organiser. Once A_2 accepts to join the team, A_1 elaborates a joint plan that enables A_1 and A_2 to resolve their conflict co-operatively and is simultaneously conflict-free with A_1's intended route. To do so, A_1 uses a modified version of the M-MRMH algorithm, which is referred to as the M-MRMH* algorithm. This version of the M-MRMH algorithm is designed for cases in which the team formed in a conflict scenario does not include all the aircraft involved in that scenario. The M-MRMH* algorithm, which is applied taking into account all the aircraft in the scenario regardless of whether they are team members or not, considers only candidate plans in which the resolution trajectories assigned to the aircraft that are



(a)



(b)

Figure 4.24. Conflict scenario 5. (a) Initial configuration (b) Predicted distances between the aircraft as they fly along their initially intended routes

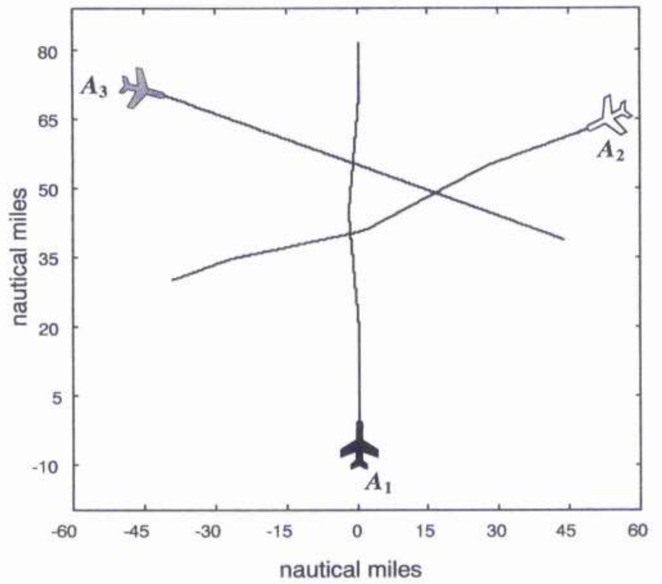
not part of the team are their respective intended trajectories. Thus, the M-MRMH* algorithm resolves the same minimisation problem as the MRMH algorithm with an additional constraint requiring the joint plans to include the intended trajectories of the aircraft that are not part of the team. In conflict scenario 5, this implies that the resolution trajectories assigned to A_1 and A_2 are guaranteed to be conflict-free with A_3's intended trajectory.

A version of the M-MRMH* algorithm adapted to the type of two-dimensional three-aircraft conflicts considered here has been implemented by appropriately modifying the implemented version of the M-MRMH algorithm. The resolution trajectories obtained from a run of the resulting version of the M-MRMH* algorithm for conflict scenario 5 are shown in Figure 4.25, together with the predicted distances between the aircraft as they fly those resolution trajectories. Table 4.9 displays the parameters defining the three resolution trajectories as well as the cost associated to each of them and to their sum.

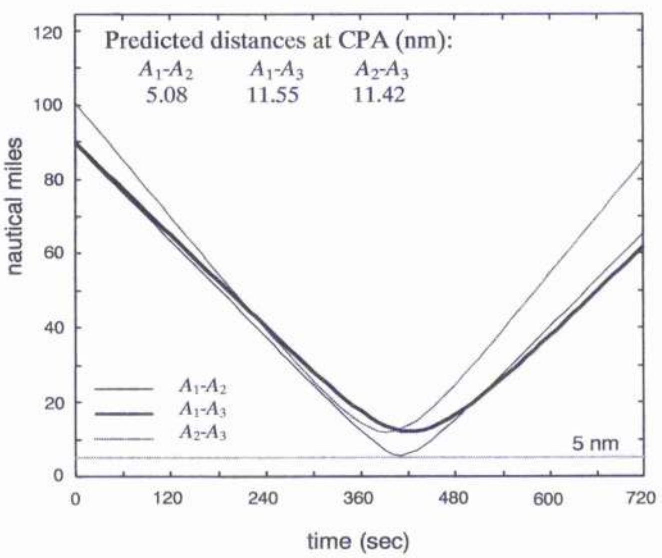
4.5.6.4 Application of reflective the co-operation mechanism conflict scenarios 6, 7 and 8

When applying the behaviouristic co-operation mechanism, conflict scenarios 6,7 and 8 are considered as type II scenarios. In this type of scenarios, two of the three aircraft are initially outside ADS-B coverage of each other. The initial positions of the aircraft in type II scenarios correspond to the time from which one of the aircraft is simultaneously within the ADS-B range of coverage of the other two. Hence, the initial positions of the aircraft when applying in conflict scenarios 6,7 and 8 are the same as when applying the behaviouristic one. The application of the reflective co-operation mechanism in these scenarios ensures that the aircraft with complete knowledge of the conflicting situation acts as the team organiser and elaborates a joint plan that resolves all the conflicts in which the aircraft are involved. Thus, the distinction between type I and type II conflict scenarios is irrelevant when applying the reflective co-operation mechanism. The aircraft that first becomes aware of the presence of the other two teams up with them regardless of whether or not they are initially within each other's ADS-B range of coverage. Hence, the reflective co-operation mechanism is applied in conflict scenarios 6, 7 and 8 analogously to how it is applied in conflict scenarios 3 and 4.

The initial configuration of the aircraft in conflict scenario 6 is reproduced in Figure 4.26. The resolution trajectories that result from a run of the M-MRMH algorithm for this scenario are shown in Figure 4.27, together with the predicted distances between the aircraft as they fly those resolution trajectories. Figure 4.28 shows a sequence of



(a)



(b)

Figure 4.25. Example of the application of the planning algorithm in scenario 5. A_3 is not involved in any conflict. A_1 acts as the team organiser of the team formed by itself and A_2. A_1 applies the M-MRMH algorithm considering the three aircraft and imposing that A_3's resolution trajectory in the resulting must be its initially intended route. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

snapshots of the positions of the aircraft along the resolution trajectories. Table 4.10 displays the parameters defining the three resolution trajectories as well as the cost associated to each of them and their sum.

	γ	t_a	t_c	**Cost**
A_1	-5°	214 s	616 s	1.44
A_2	-8°	204 s	629 s	2.59
A_3	Initially intended route			0.00
Total conflict resolution cost				4.03

Table 4.9. Example of the application of the planning algorithm in scenario 5. A_1 acts as the team organiser. Values of parameters and costs of the resolution trajectories in Figure 4.25(a)

To obtain samples of the random variables c_1, c_2, c_3, s_c, $d_{A1\text{-}A2}$, $d_{A1\text{-}A3}$ and $d_{A2\text{-}A3}$ for this scenario, the M-MRMH algorithm has been run 50 times. The samples reflect the probabilistic character of the algorithm and are drawn according to the PDFs $f(c_1)$, $f(c_2)$, $f(s_c)$, and $f(d_{A1\text{-}A2})$, respectively. The means of the samples obtained are displayed in Table 4.11. This table shows that the application of the reflective co-operation mechanism in this scenario results in A_1 maintaining its initially intended route and A_2 and A_3 sharing the total resolution cost in an equitable manner.

To identify the differences between the performance of the two co-operation mechanisms in conflict scenario 6, Table 4.11 is compared with Table 3.19, which displays the sample means resulting from the repeated simulation of the application of the behaviouristic mechanism in this conflict scenario assuming that A_1 acts as the leader. The comparison shows that the total resolution cost is higher with the behaviouristic mechanism than with the reflective one. It is observed that, while the application of the reflective co-operation mechanism results in A_1 not manoeuvring and A_2 and A_3 sharing the total resolution cost, the behaviouristic one results in A_2 not manoeuvring and A_1 bearing most of the total resolution cost.

The M-MRMH algorithm has been repeatedly run 50 times for conflict scenarios 7 and 8. These conflict scenarios represent the same conflicting configuration as conflict scenario 6 with an ADS-B range of coverage of 90 nm and 100 nm, respectively, instead of 80 nm. The initial positions of the aircraft in these scenarios are shown, respectively, in Figures 3.52 and 3.53. The means of the samples obtained for scenario 7 are shown in Table 4.12 and the means of the samples obtained for scenario 8 are shown in Table 4.13.

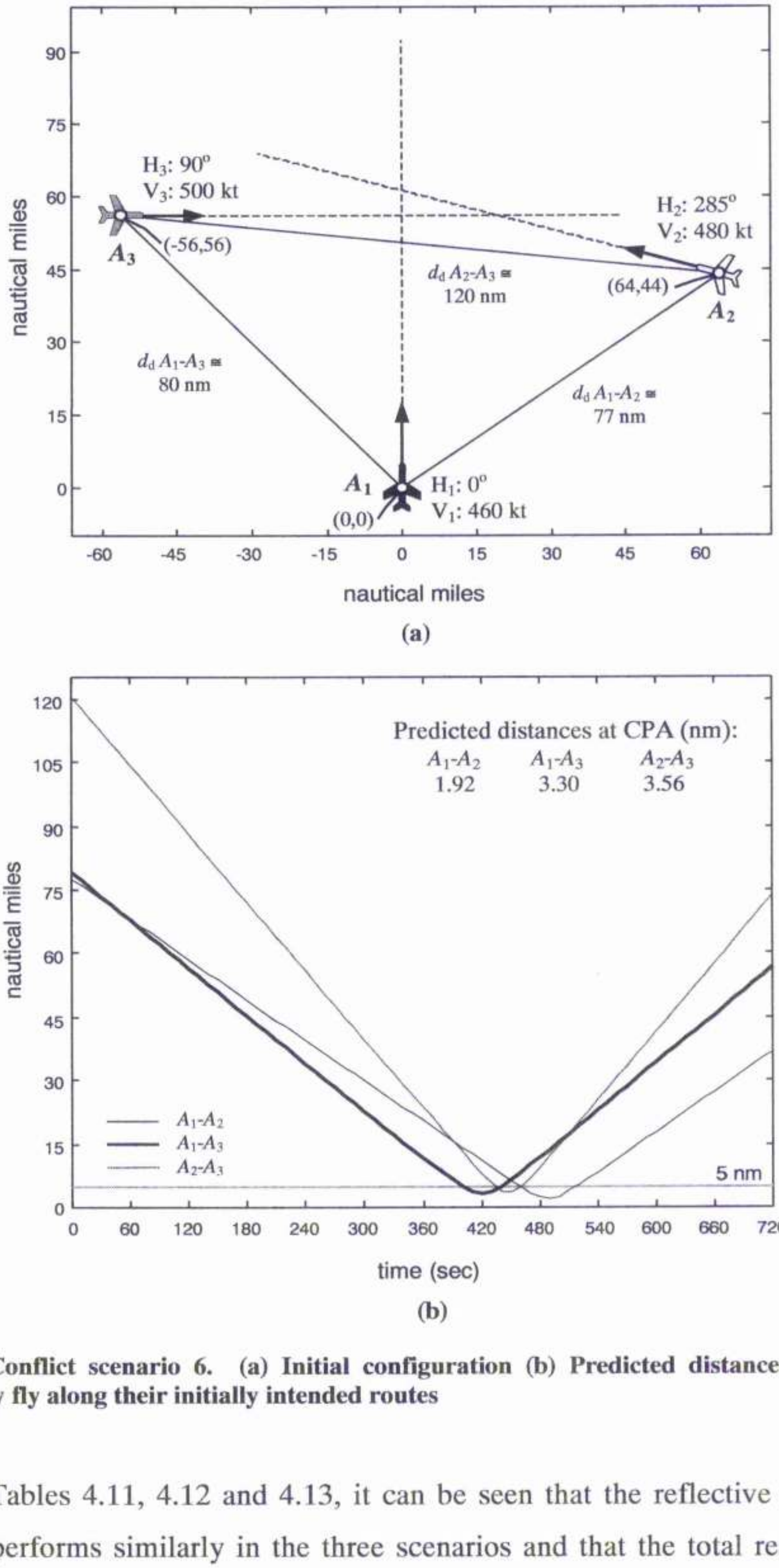
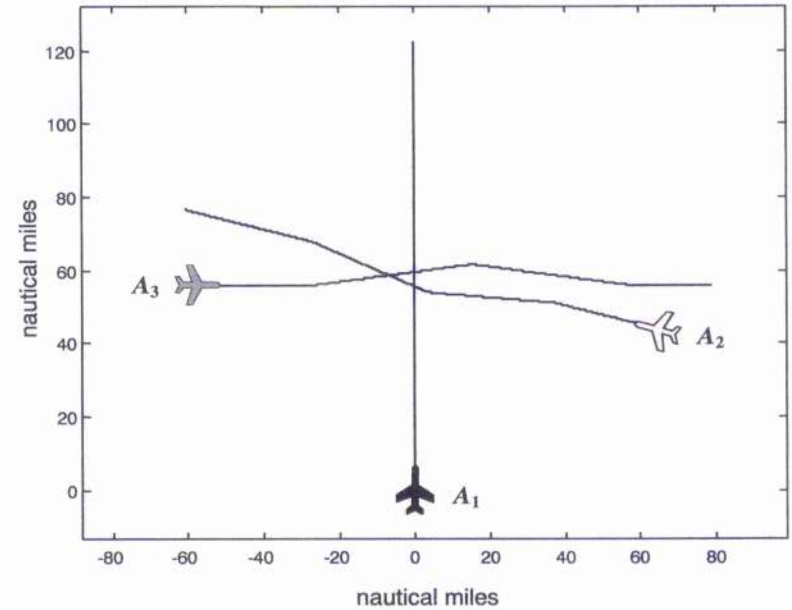
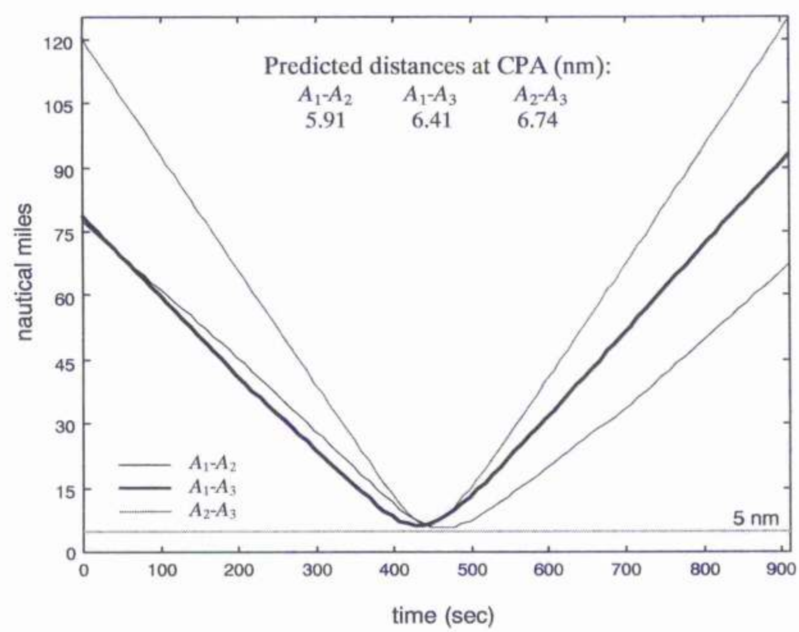


Figure 4.26. Conflict scenario 6. (a) Initial configuration (b) Predicted distances between the aircraft as they fly along their initially intended routes

Comparing Tables 4.11, 4.12 and 4.13, it can be seen that the reflective co-operation mechanism performs similarly in the three scenarios and that the total resolution cost decreases when the ADS-B range of coverage increases. In the three conflict scenarios the reflective co-operation mechanism results in A_1 not manoeuvring and A_2 and A_3



(a)



(b)

Figure 4.27. Example of the application of the planning algorithm in scenario 6. A_2 and A_3 are initially outside the ADS-B coverage of each other. A_1 is the team organiser. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

sharing the total resolution cost. It can also be observed that the smaller the ADS-B range of coverage, the more evenly the total resolution cost is distributed between A_2 and A_3.

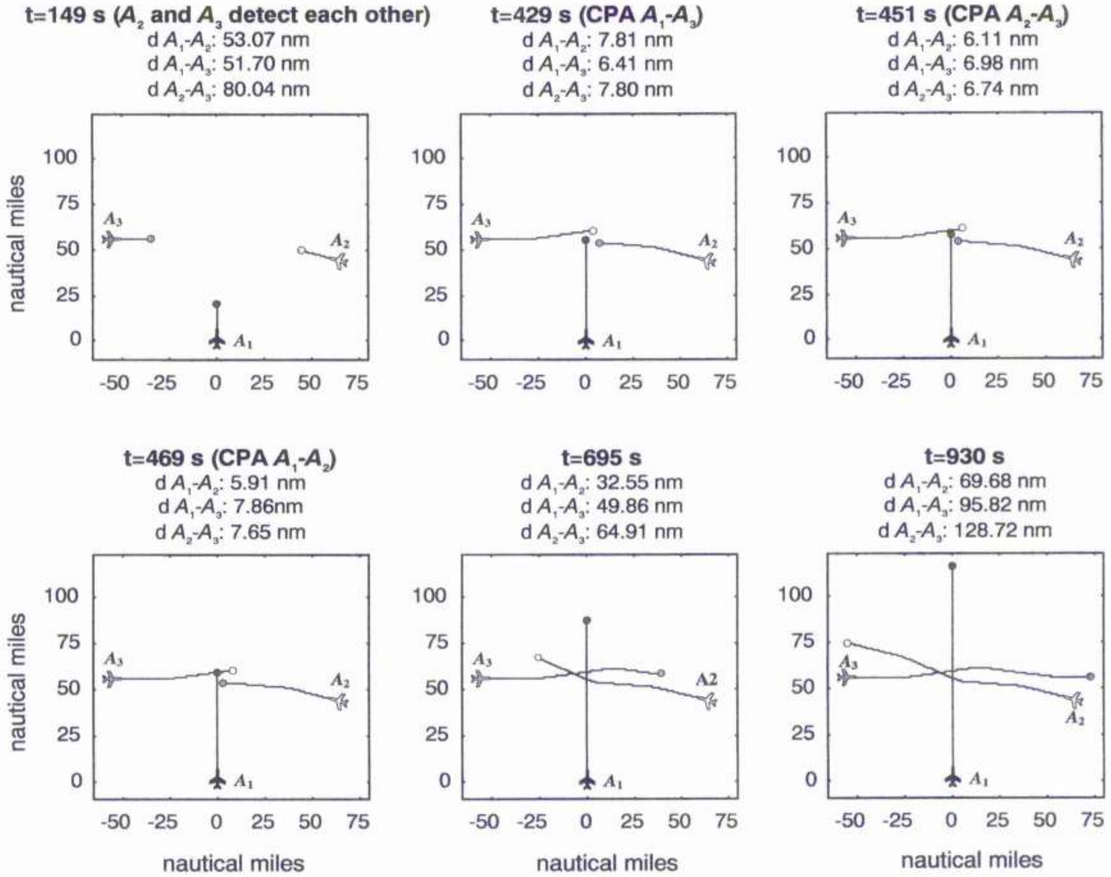


Figure 4.28. Example of the application of the planning algorithm in scenario 6: sequence of predicted future positions of the aircraft along their resolution trajectories in Figure 4.27(a)

	γ	t_s	t_e	**Cost**
A_1	Initially intended route			0.00
A_2	-10^o	213 s	711 s	3.51
A_3	-8^o	204 s	827 s	2.84
Total conflict resolution cost				6.35

Table 4.10. Example of the application of the planning algorithm in scenario 6. Values of parameters and costs of the resolution trajectories in Figure 4.27(a)

To identify the differences between the performance of the two co-operation mechanisms in conflict scenarios 7 and 8, Tables 4.12 and 4.13 are compared, respectively, with Table 3.20 and 3.21. Tables 3.20 and 3.21 display the sample means resulting from the repeated simulation of the application of the behaviouristic mechanism assuming that A_1 acts as the leader in conflict scenarios 7 and 8,

$\bar{x}$ Cost	A_1	0.00
	A_2	3.26
	A_3	2.77
	Total	6.03
$\bar{x}$ CPA distance (nm)	A_1-A_2	5.51
	A_1-A_3	6.09
	A_2-A_3	5.76

Table 4.11. Sample means of the resolution costs and the distances between the aircraft at their CPA for samples obtained simulating 50 times the application of the planning algorithm in conflict scenario 6

$\bar{x}$ Cost	A_1	0.00
	A_2	3.21
	A_3	1.95
	Total	5.16
$\bar{x}$ CPA distance (nm)	A_1-A_2	5.37
	A_1-A_3	5.87
	A_2-A_3	5.63

Table 4.12. Sample means of the resolution costs and the distances between the aircraft at their CPA for samples obtained simulating 50 times the application of the planning algorithm in conflict scenario 7

$\bar{x}$ Cost	A_1	0.00
	A_2	3.20
	A_3	1.79
	Total	4.99
$\bar{x}$ CPA distance (nm)	A_1-A_2	5.30
	A_1-A_3	6.07
	A_2-A_3	6.25

Table 4.13. Sample means of the resolution costs and the distances between the aircraft at their CPA for samples obtained simulating 50 times the application of the planning algorithm in conflict scenario 8

respectively. The comparison between Table 4.12 and 3.20 leads to the same conclusions as those reached for conflict scenario 6. However, that is not the case for conflict scenario 8, in which the behaviouristic co-operation mechanism performs differently than in scenarios 6 and 7. As explained in the previous chapter, the application of the behaviouristic mechanism in this scenario generally results in the three aircraft manoeuvring despite the fact that A_1 plans its resolution trajectory with the intention of allowing A_3 to maintain its initially intended route. This can be explained by the large ADS-B range of coverage in this conflict scenario, which makes it difficult for A_1 to accurately predict the other aircraft's reactions to its resolution trajectory. As a consequence of the three aircraft having to manoeuvre, the behaviouristic mechanism results in the three aircraft sharing the total resolution cost equitably, as shown in Table 3.21. On the other hand, the reflective co-operation mechanism results in A_1 not manoeuvring and A_2 and A_3 sharing the total resolution cost. Although the total resolution cost is lower with the reflective mechanism than with the behaviouristic one, the latter results in a more even distribution of that total cost among the three aircraft than the former.

4.5.6.5 Application of the reflective co-operation mechanism conflict scenario 9: example of even distribution of the total cost among the three aircraft

The application of the reflective co-operation mechanism in the three-aircraft conflict scenarios considered so far has been shown to result in one of the three aircraft maintaining its initially intended trajectory and the other two sharing equitably the total resolution cost. In other conflicting configurations, the M-MRMH algorithm may produce joint plans that distribute the total resolution cost evenly amongst the three aircraft. An example of such a conflicting configuration is conflict scenario 9, which is depicted in Figure 4.29. The ADS-B range of coverage in this scenario is assumed to be 90 nm. As it can be seen in Figure 4.29(b), each of the three aircraft will conflict with the other two if they maintain their currently intended trajectories. It can be observed that this conflict scenario is very similar to conflict scenario 4. The aircraft speeds and heading are the same in both scenarios and the aircraft initial positions differ in only a few nautical miles.

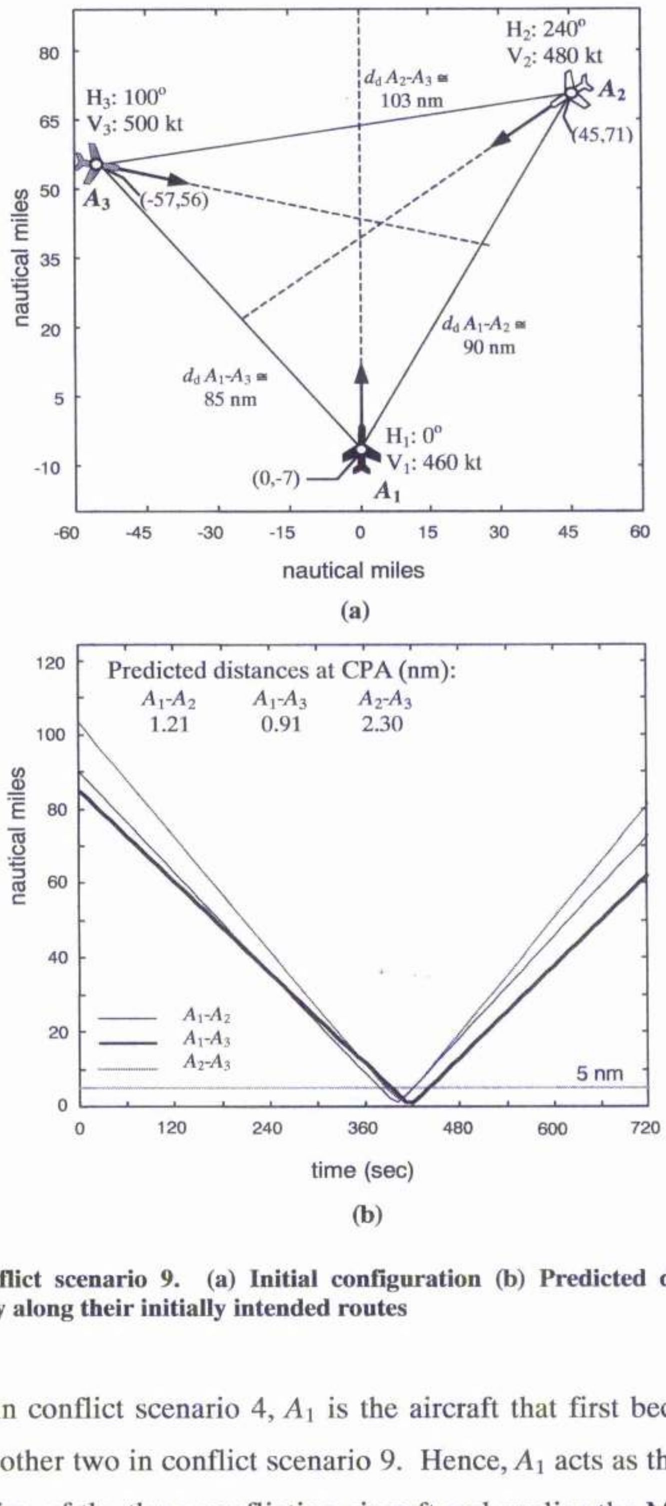


Figure 4.29. Conflict scenario 9. (a) Initial configuration (b) Predicted distances between the aircraft as they fly along their initially intended routes

As is the case in conflict scenario 4, A_1 is the aircraft that first becomes aware of the presence of the other two in conflict scenario 9. Hence, A_1 acts as the team organiser of a team comprising of the three conflicting aircraft and applies the M-MRMH algorithm to elaborate a joint conflict resolution plan. The resolution trajectories that form the joint plan obtained from a run of the algorithm are shown in Figure 4.30 together with the predicted distances between the aircraft as they fly those resolution trajectories. In this case, the joint plan requires the three aircraft to manoeuvre. Table 4.14 displays the parameters defining the three resolution trajectories in Figure 4.30(a) as well as the cost associated to each of them and the total resolution cost. Table 4.15 shows the means of the samples from the random variables $\boldsymbol{c_1}$, $\boldsymbol{c_2}$, $\boldsymbol{c_3}$, $\boldsymbol{s_c}$, $\boldsymbol{d_{A1\text{-}A2}}$, $\boldsymbol{d_{A1\text{-}A3}}$ and $\boldsymbol{d_{A2\text{-}A3}}$ drawn by repeatedly running 50 times the algorithm in this scenario.

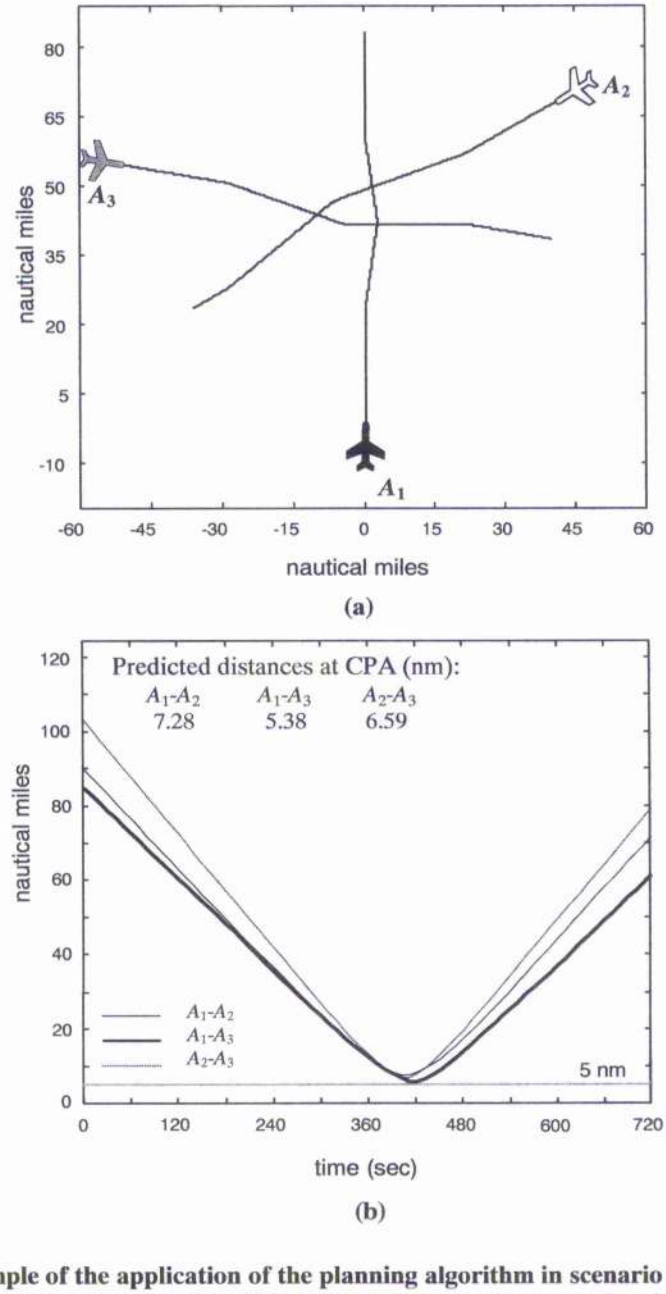


Figure 4.30. Example of the application of the planning algorithm in scenario 9. A_1 acts as the team organiser. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

	γ	t_s	t_e	Cost
A_1	8^0	240 s	539 s	2.37
A_2	10^0	201 s	656 s	3.42
A_3	10^0	201 s	591 s	3.32
Total conflict resolution cost				9.11

Table 4.14. Example of the application of the planning algorithm in scenario 9. A_1 acts as the team organiser. Values of parameters and costs for the resolution trajectories in Figure 4.30(a)

$\bar{x}$ Cost	A_1	3.00
	A_2	3.14
	A_3	3.26
	Total	9.41
$\bar{x}$ CPA distance (nm)	A_1-A_2	7.40
	A_1-A_3	6.41
	A_2-A_3	6.49

Table 4.15. Sample means of the resolution costs and the distances between the aircraft at their CPA for samples obtained simulating 50 times the application of the planning algorithm in conflict scenario 9

Comparing Table 4.15 with Table 4.8, which corresponds to conflict scenario 4, it can be observed that, despite the fact that the two conflict scenarios represent very similar conflicting configurations, the M-MRMH algorithm results in a more even distribution of the total resolution cost among the three aircraft in conflict scenario 9 than in conflict scenario 4. However, the total resolution cost is higher in conflict scenario 9 than in conflict scenario 4.

4.5.6.6 Application of the reflective co-operation mechanism in non-nominal situations

In this section, the M-MRMH* algorithm is shown to be applicable in no-nominal situations where one of the conflicting aircraft does not join the team proposed by the team organiser. Throughout this chapter it has been assumed that, in nominal operation, a conflicting aircraft-agent is always willing to join a team to resolve the conflicts in which it is involved. When an aircraft-agent decides to join a particular team, its flight crew is nominally assumed to accept it. However, in the event of a non-nominal occurrence such as equipment failure or distress in the cockpit, an aircraft-agent, or its flight crew, may reject to join a team and unilaterally decide to pursue the route that is most appropriate for the aircraft given the circumstances. When such a situation arises, the team organiser can still team up with the remaining conflicting aircraft-agents and use the M-MRMH* algorithm to elaborate a joint plan that is conflict-free with the intended trajectory of the aircraft-agent that does not join the team. The M-MRMH* algorithm, which is applied taking into account not only the team members but also the aircraft-agents that are not part of the team, considers only the candidate plans in which the resolution trajectories assigned to the aircraft-agents that do not join the team are their respective intended trajectories. This implies that the conflict-free joint plan that results of running the M-MRMH* algorithm includes those trajectories.

To illustrate the above with an example, the application of the reflective co-operation mechanism in conflict scenario 6 is simulated assuming that a non-nominal occurrence on board A_3 prevents it from joining the team proposed by the team organiser A_1. Suppose that A_2 accepts A_1's team membership proposal but A_3 rejects it and decides to maintain its initially intended route. Thus, the team formed by A_1 includes only A_1 and A_2. In this situation, A_1 uses the M-MRMH* algorithm to elaborate a conflict-free joint plan that includes A_3's intended trajectory and enables A_1 and A_2 to share the total cost of resolving all the conflicts in the scenario. The resolution trajectories obtained from a run of the M-MRMH* algorithm for conflict scenario 6 are shown in Figure 4.31 together with the predicted distances between the aircraft as they fly those resolution trajectories. Table 4.16 displays the parameters defining the three resolution trajectories as well as the cost associated to each of them and to their sum. In this table it can be seen that the total resolution cost is evenly distributed between A_1 and A_2.

4.6 Conclusions

This chapter has illustrated the potential use of multi-agent systems concepts and techniques to develop co-operative conflict resolution methodologies for Autonomous Aircraft equipped with an air-to-air data-link. A reflective co-operation mechanism has

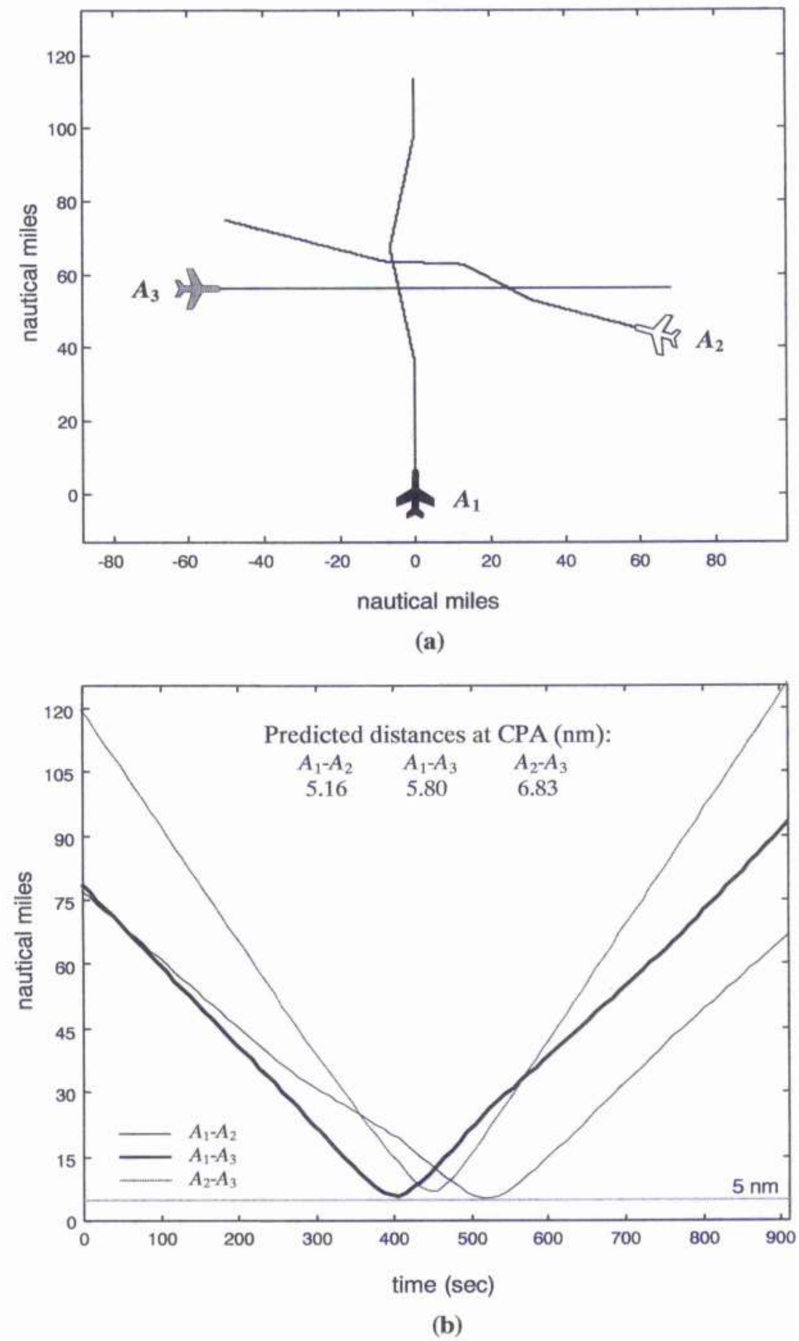


Figure 4.31. Example of the application of the planning algorithm in scenario 6 assuming that A_3 does not join the team. A_1 is the team organiser. (a) Resolution trajectories (b) Predicted distances between the aircraft as they fly their resolution trajectories

been proposed to enable conflicting Autonomous Aircraft in Operational Environment B to associate in teams so that they can co-ordinate their conflict resolution actions and share the resolution costs. Conflicting Autonomous Aircraft have been modelled as

	γ	t_s	t_e	Cost
A_1	-12°	286 s	772 s	4.36
A_2	13°	257 s	567 s	4.31
A_3	Initially intended route			0.00
Total conflict resolution cost				8.67

Table 4.16. Example of the application of the planning algorithm in scenario 6 assuming that A_3 does not join the team. A_1 acts as the team organiser. Values of parameters and costs for the resolution trajectories in Figure 4.31(a)

reflective agents within a multi-agent system. According to this model, the aircraft-agents interact with one another to resolve their conflicts co-operatively. Co-operation has been viewed from the reflective perspective, according to which the aircraft-agents are deemed to co-operate when they are knowingly committed to resolve the conflicts in which they are involved through the implementation of an agreed joint plan. In principle, either the reflective or the behaviouristic perspective could have been adopted in this operational environment. Since the potential of the behaviouristic perspective was explored in chapter 3 for Operational Environment A, the reflective one has been adopted in this case to provide a more complete picture of the possibilities of the proposed multi-agent approach to conflict resolution in AAO.

The application of the proposed reflective co-operation mechanism results in one of the conflicting aircraft-agents, referred to as the team organiser, forming a team that encompasses all the aircraft-agents in the vicinity involved in at least one conflict. The team members are committed to the resolution of all the conflicts in which they are involved through the implementation of a joint plan elaborated by the team organiser. The joint plan includes a resolution trajectory for each of the team members. All the aircraft-agents are equipped with a planning algorithm that enables them to elaborate joint plans when they act as team organisers. In principle, each aircraft-agent may use a different planning algorithm, as long as it produces joint plans consisting of feasible resolution trajectories that result in the resolution of all the conflicts and distribute the resolution costs evenly among the team members. Once it has elaborated the joint plan, the team organiser uses the data-link to inform its team mates of their respective resolution trajectory within the joint plan. The resolution trajectories are assumed to be encoded in a standard trajectory language known to all the team members.

An example of a planning algorithm that could be used by team organisers to elaborate joint plans has been proposed. This planning algorithm, which has been named Modified Multi-start Random Mutation Hillclimbing algorithm (M-MRMH), is in fact a combinatorial optimisation algorithm based on the Random Mutation Hillclimbing meta-heuristic. The problem of searching for a set of feasible resolution trajectories that distribute the total resolution cost equitably among the team members has been recast as a multi-objective combinatorial minimisation problem involving the individual cost functions of all the team members. This multi-objective minimisation problem has been converted to a single-objective minimisation problem by applying the Linear Weighting Method. The M-MRMH algorithm elaborates a joint plan that is an approximation of a Pareto optimum of the original multi-objective minimisation problem.

To compare the performance of the reflective co-operation mechanism with the M-MRMH algorithm with that of the behaviouristic one, the algorithm has been adapted to two-dimensional conflict scenarios involving up to three aircraft. The resolution trajectories allowable to the conflicting aircraft-agents in those scenarios have been limited to lateral shift manoeuvres, as it was done for the behaviouristic mechanism. The weights of the function to be minimised have been adjusted so that the M-MRMH algorithm produces joint plans that approximately minimise the total resolution cost. Due to the probabilistic nature of the iterative improvement processes performed by the M-MRMH algorithm, its output is random and unpredictable *a priori*. Consequently, the performance of the mechanism has been analysed from a statistical perspective. In the two-dimensional conflict scenarios studied, the reflective co-operation mechanism with the M-MRMH algorithm has been shown to result, in general, in a more even distribution of the total resolution cost among the conflicting aircraft than the behaviouristic one.

The M-MRMH algorithm has been modified to make it applicable in situations where one or more of the aircraft in a conflict scenario do not join the team proposed by the team organiser. The modified version of the algorithm, which has been named the M-MRMH* algorithm, has been shown to be successfully applicable in scenarios in which one of the aircraft is not involved in any conflict and also in non-nominal situations where one of the conflicting aircraft does not join the team. In both cases, the M-MRMH* enables the two team members to resolve all the conflicts in the scenario while allowing the aircraft that does not join the team to maintain its initially intended route.

Chapter 5

Conclusions and issues for further study

5.1 Introduction

The work presented in this thesis has proposed a new approach to airborne separation assurance in Autonomous Aircraft Operations. The approach is based on considering conflicting Autonomous Aircraft as co-operating agents in the context of a multi-agent system. This multi-agent approach provides a modelling framework that facilitates the development of co-operation mechanisms allowing Autonomous Aircraft to co-ordinate their conflict resolution actions safely while sharing the conflict resolution costs fairly. Two co-operation mechanisms have been developed using the proposed multi-agent approach to illustrate its capabilities. One of the co-operation mechanisms is behaviouristic and the other is reflective. Each of them has been designed for a specific operational environment and can be seen as a co-operative conflict resolution methodology for the operational concept of AAO adopted in this thesis. This chapter discusses the main conclusions drawn from the analysis of the two mechanisms and makes recommendations for further research regarding each of them.

5.2 Behaviouristic co-operation mechanism: conclusions and recommendations for further research

The performance analysis of the behaviouristic co-operation mechanism in two-dimensional conflict scenarios showed that it always results in conflict-free resolution strategies in the scenarios considered. The ability of the mechanism to produce conflict-free resolution strategies is a consequence of the following three features:

- The leader-follower conflict resolution scheme, which dictates that an aircraft's resolution trajectory must be conflict-free with the resolution trajectories of its conflicting leaders.
- The ability of the trajectory-planning algorithm to gradually increase the value of γ_p until at least one conflict-free combination of patterns is found.
- The capacity of the aircraft-agent with complete knowledge of the conflicting situation in type II conflict scenarios to impose its leadership on the other two aircraft.

Further analysis is required to prove whether or not these three features are sufficient to ensure conflict-free strategies in any two-dimensional conflicting configuration involving up to three aircraft.

In the scenarios investigated, the co-operation mechanism has been shown to enable the conflicting aircraft to share the costs of resolving the conflicts in which they are involved. The distribution of the resolution costs among the conflicting aircraft varies with the conflicting configuration.

The co-operation mechanism has been designed to assist the flight crews of Autonomous Aircraft in resolving conflicts. The application of the mechanism requires minimal human intervention. In nominal operations, the flight crew's tasks are simply to understand the resolution trajectory proposed by the algorithm and to accept it as the aircraft's new intended trajectory. Consequently, the involvement of the flight crew in the conflict resolution process could be considered excessively low. A higher degree of interaction between the flight crew and the ASAS equipment would be a step towards increasing the flight crew's level of involvement in the conflict resolution process. For example, the co-operation mechanism could be modified so that the flight crew could input the efficiency criteria to be taken into account by the cost function and request additional runs of the algorithm to widen the choice of possible resolution trajectories.

The application of the co-operation mechanism requires a high degree of trust in the ASAS equipment as the flight crew are required to accept the proposed resolution trajectory even if it is not conflict-free with the currently intended trajectories of the conflicting followers. During the course of the conflict resolution process the flight crew might be offered successive resolution trajectories that are consecutively cancelled. This may be disconcerting for the crew. It can be argued that training and experience will enhance trust in the co-operation mechanism. Nevertheless, further research is required to analyse the co-operation mechanism from a human factors standpoint.

Further research should also focus on extending the trajectory-planning algorithm so that it can be applied in generic three-dimensional conflict scenarios. New types of resolution trajectories should be introduced, together with conflict resolution patterns representing those types. The extended algorithm should be able to anticipate the types of resolution trajectories that the conflicting aircraft may fly along, in any possible conflicting configuration. The limitations of the extended algorithm regarding the maximum number of conflicting aircraft it can handle must be explored. The ground-based ATFM service would have to consider those limitations to regulate the air traffic density in AAO airspace in order to ensure that all potential conflicts are resolvable by the Autonomous Aircraft.

Once the trajectory-planning algorithm has been extended to generic three-dimensional conflicts, an agent program enabling the practical implementation of the co-operation mechanism should be developed. Rule-based and object-oriented software techniques could be explored as methodologies to implement the agent program. The trajectory-planning algorithm should be integrated into the agent program, which would run on an on-board computer. Conflicting aircraft-agents would execute their agent program to apply the co-operation mechanism. Once a suitable agent program has been developed, the application of the co-operation mechanism could be realistically simulated.

Another interesting topic for further research is the extension of the co-operation mechanism to handle non-nominal situations. This research would take into consideration human factors issues.

5.3 Reflective co-operation mechanism: conclusions and recommendations for further research

The performance analysis of the reflective co-operation mechanism with the M-MRMH algorithm in two-dimensional conflicts showed that it enables the resolution of all the conflicts in the scenarios considered. The following two features result in the mechanism's ability to produce conflict-free resolution strategies:

- The protocol for team formation establishes that the aircraft-agents must wait for a certain time period, t_f, before joining a team. This ensures that the team members of a formed team will only be involved in conflicts with other team members.
- The M-MRMH algorithm gradually increases the value of γ_p until at least one conflict-free plan-pattern is found.

In the two-dimensional scenarios considered, the value of t_f was 30 seconds. Considering t_f, together with the model of flight crew response latency adopted, it was assumed that a team encompassing all the aircraft in the scenario was formed within 200 seconds of the corresponding team membership proposal being issued. This assumption implied that, within those 200 seconds, all the flight crews must first have become aware that the team was being formed and have accepted to join it. Subsequently, they must have understood their resolution trajectories and have agreed to their implementation. As was the case for the behaviouristic mechanism, the application of the reflective co-operation mechanism required a low level of involvement of the flight crew in the resolution process and a high degree of trust in the ASAS equipment. The analysis of the human factors issues arising from the role of the flight crew in the conflict resolution process could be the subject of future research.

Further research is also required in order to prove whether or not the M-MRMH algorithm is capable of finding at least one conflict-free plan-pattern in any two-dimensional conflicting configuration involving up to three aircraft.

In the two-dimensional conflict scenarios studied, the reflective co-operation mechanism was generally shown to result in a more even distribution of the total resolution costs among the conflicting aircraft compared to the behaviouristic mechanism. This is justified by the fact that the application of the reflective co-operation mechanism results in the conflicting aircraft-agents forming a team so that one of them, the team organiser, can plan the resolution trajectories of all the members of the team. On the contrary, the application of the behaviouristic mechanism requires each conflicting aircraft-agent to compute its own resolution trajectory by considering the possible responses of other aircraft-agents to that trajectory. Even though the aircraft-agents are willing to collaborate in resolving conflicts, the lack of means available to agree on a joint strategy may prevent the aircraft-agents from sharing the resolution costs equitably.

The reflective mechanism's ability to distribute the costs evenly amongst the conflicting aircraft is achieved at the expense of a very high communications overhead. Whilst the behaviouristic co-operation mechanism relies only on ADS-B, the reflective mechanism also requires a high volume of one-to-one communications so that a team can be formed and its team organiser can inform the team members of their respective resolution trajectories. The extensive use of one-to-one communications results in only one of the conflicting aircraft employing its computational resources, while the behaviouristic mechanism requires all the conflicting aircraft to do so.

Further research regarding the reflective co-operation mechanism should include the extension of the M-MRMH to generic three-dimensional conflict scenarios. To achieve this, new categories of resolution trajectories should be introduced. The limitations of the extended algorithm regarding the maximum number of aircraft it can handle should also be explored. These limitations will have to be taken into account by the ground-based ATFM service to regulate the air traffic density in AAO airspace. The possible conflict configurations that may arise should be analysed in order to select a value of t_f that prevents conflicts with aircraft that are not part of the team. In addition, conflicting situations that may require the successive application of the co-operation mechanism should be investigated. In such situations, the members of a team should be able to determine when to safely drop their commitment to their current team so that they can join a new one to resolve other conflicts.

An agent program enabling the practical implementation of the reflective co-operation mechanism should also be developed. Rule-based and object-oriented software techniques could be explored as methodologies to implement the agent program. The team formation protocol should be integrated into the agent program as well as the extended M-MRMH algorithm. Once a suitable agent program has been developed, the application of the co-operation mechanism could be realistically simulated.

Another possible topic for further research is the extension of the reflective co-operation mechanism's ability to handle non-nominal situations. Research on this topic would take into consideration human factor issues. The M-MRMH* algorithm, a version of the M-MRMH algorithm applicable in situations where not all the conflicting aircraft join the team, could be considered as a first step towards a reflective mechanism capable of coping with non-nominal situations.

5.4 Comparison with other approaches to conflict resolution for Autonomous Aircraft

As opposed to most of the previous work on conflict resolution in AAO, the two conflict resolution methodologies proposed in this thesis rely on long-term intent information to detect conflicts. This allows for early conflict detection, which in turn enables the proposed algorithms for planning resolution trajectories to take into account cost efficiency as well as safety. While planning algorithms that consider the cost of the resolution trajectories has been proposed elsewhere, an operational framework within which those algorithms could be implemented has not been provided. The methodologies for conflict resolution in AAO presented in this thesis comprise of not only planning algorithms but also the necessary operational procedures for the flight crew to apply them. In addition, the proposed conflict resolution methodologies allow Autonomous Aircraft to resolve conflicts in a safe and co-ordinated manner while sharing the resolution costs. Previous work on conflict resolution for self-separating has focused on the co-ordination of the aircraft's manoeuvres without considering explicitly the distribution of the resolution costs among the conflicting aircraft. One of the advantages of the proposed conflict resolution methodologies over some of the previously proposed ones is that the computed resolution trajectories are made of standard manoeuvres that are readily understood by pilots and compatible with the FMS.

The behaviouristic co-operation mechanism presented in this thesis is based on information broadcast through ADS-B, as it is the case for the previously proposed schemes for conflict resolution in AAO. On the other hand, the reflective co-operation mechanism constitutes the first conflict resolution methodology proposed to date based on a point-to-point data-link. The use of this data-link facility enables one of the conflicting aircraft to acquire a global picture of the conflicting situation, which is not guaranteed in ADS-B-based conflict resolution methodologies.

Appendix A

The Airborne Collision Avoidance System (ACAS)

A.1 Historical background of ACAS

As the result of a joint meeting held in 1955, the Radio Technical Commission for Aeronautics (RTCA), the Institution of Electrical and Radio Engineers (IERE) and the Air Transport Association (ATA) issued a formal request to the electronics industry. They called for the development of an on board system that contributed to reduce the increasing threat of mid-air collisions. Such a system would complement the ground-based Air Traffic Control Services, particularly in airspace that was outside the areas of radar coverage. The development of a capable and affordable airborne collision avoidance system intensified when in 1956 two airliners collided over the Grand Canyon region of the United States. It was felt that, with the continual increase of air traffic, the development of a system able to provide mid-air collision avoidance independently of the aircraft navigation equipment and the ground control was needed.

Several experimental airborne warning devices were developed during the 1950s and 1960s, but they were considered impractical due either to their high costs or to their inability to deliver an adequate warning to the pilot. One of these devices was the Eliminate Range-zerO System (EROS), designed for fast moving, fighter-type aircraft.

EROS used time-frequency techniques to determine the range to proximate aircraft as well as their relative speed [108]. EROS was considered impractical due to its high cost and was never used. In 1958 John S Morrel published a study about collision avoidance in which he proposed an alerting system based on the time-to-go to the closest point of approach [109]. This work is one of the most important milestones in the road to the current ACAS concept.

By the late 1960s, the United States had a wide ground-based Secondary Surveillance Radar (SSR) coverage that gave air traffic controllers the location and altitude of Mode C transponder-equipped aircraft. The availability of this radar information enabled controllers to alert pilots about possible conflicts with nearby traffic. As a result of both this new air traffic environment and the diverse experimental devices developed during the previous years, a system called Beacon Collision Avoidance System (BCAS) was devised. BCAS relied on the timing of transponder replies of proximate aircraft to ground interrogators [108]. BCAS never went into full production because it was considered too complex and would not work over the ocean or where there was limited SSR coverage.

In 1981, the Federal Aviation Administration decided to pursue a fully onboard design approach to collision avoidance and the Traffic Collision Avoidance System (TCAS) concept was introduced. While the TCAS project was being developed, ICAO was developing the concept of the Airborne Collision Avoidance System (ACAS), which was finally introduced in November 1993. Besides the definition of the ACAS concept, ICAO elaborated the set of standards, regulations and recommended practices which commercial implementations of ACAS would have to comply with ([110], [111], [112]). Thus, ACAS is the result of ICAO's efforts to define an airborne system liable to be accepted by the aviation community as the last recourse to prevent mid-air collisions. Hence, ACAS operates independently of the aircraft navigation equipment and the ground-based ATM. The ACAS concept comprises the provision of several services to the pilot, such as enhanced traffic situation awareness, traffic proximity warning, imminent collision alert and recommended collision avoidance manoeuvres. To provide these services, the ACAS equipment tracks the proximate aircraft by interrogating the Secondary Surveillance Radar (SSR) transponders installed on them.

In the Annex 2 to the Convention on International Civil Aviation, dedicated to the Rules of the Air [12], ICAO defines three types of ACAS: ACAS I, ACAS II and ACAS III. They are briefly outlined below.

- **ACAS I** locates aircraft in the immediate vicinity and displays their location to the pilot. ACAS I is unable to compute avoidance courses and therefore it does not provide the pilot with recommended resolution manoeuvres. ACAS I issues a type of alert to the pilot called Traffic Advisory (TA), which warns the pilot of potentially threatening proximate aircraft.

- **ACAS II** constantly interrogates the surrounding transponder-equipped aircraft to acquire their relative positions, speeds and altitudes. The system is equipped with a collision avoidance logic that enables it to issue Resolution Advisories (RAs) in addition to TAs. RAs are alerts in which a recommended collision avoidance manoeuvre is issued to the pilot. ACAS II only provides RAs in the vertical plane.

- **ACAS III** provides the pilot with TAs and RAs in both the vertical and horizontal planes.

As far as the actual implementation of ACAS is concerned, only TCAS, currently built by three different North American manufacturers, complies with the standards elaborated by ICAO. Thus, TCAS I complies with ACAS I standards and TCAS II with ACAS II standards. No ACAS III equipment has been developed to date and none is likely to appear in the near future due to technical and operational complications.

In the United States, carriage of TCAS II equipment has been mandatory for aircraft seating more than 30 passengers since the 30th of December 1993 [113]. In 1995, the EUROCONTROL Committee of Management approved an implementation policy and schedule for the mandatory carriage of ACAS II in Europe. According to this schedule, all aircraft with a maximum taking-off mass exceeding 15,000 kg or carrying more than 30 passengers flying in the airspace of any of the European Civil Aviation Conference member states are required to install ACAS II from March 2001. The Version 7 of the TCAS II equipment is required to fully comply with the ACAS II Standard and

Recommended Practices published by ICAO [112]. Mandatory carriage of ACAS II is also being implemented in many other states such as Argentina, Australia, Chile, Egypt, India, Japan, etc.

A.2 Implementation of ACAS II: TCAS II Version 7

The fully operative airborne collision avoidance system complying with the ACAS II Standard And Recommended Practices published by ICAO [112] is TCAS II Version 7. In this section TCAS II Version 7 will be described and its main components and features will be outlined.

A.2.1 System components

The main elements of the TCAS II Version 7 equipment are depicted in Figure A.1 and are briefly explained below:

- Computer unit, which performs target tracking and implements the collision avoidance logic to detect conflicts and generate advisories.
- Interrogator, which operates on the standard SSR uplink frequency of 1030 MHz. The replies are received on 1090 MHz, the downlink SSR frequency.
- Two antennas, one of them directional and fitted at the top of the fuselage and the other omni-directional and placed at the bottom of the fuselage.
- Connection with the Mode S transponder, to co-ordinate resolution advisories with TCAS II Version 7-equipped conflicting aircraft.
- Connection with the barometric altimeter or with Air Data Computer if fitted, to obtain the pressure altitude.
- Connection with the radar altimeter, to obtain the range to the terrain and enable the TCAS system to both inhibit the issue of RAs when the aircraft is proximate to the ground and determine whether proximate aircraft being tracked are actually on the ground.
- TCAS control panel.
- Loudspeakers, for aural messages.

- Screen, to display the TCAS related data and the Traffic and Resolution Advisories.

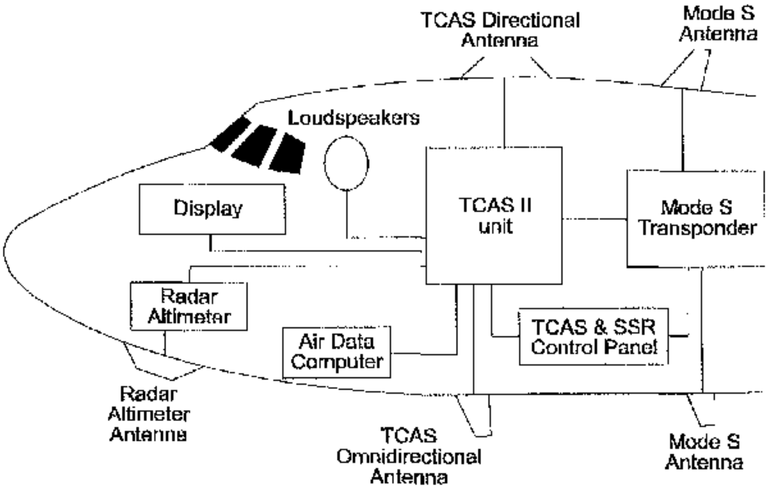


Figure A.1: Main elements of the TCAS II Version 7 equipment (ACAS brochure [114]).

TCAS II Version 7 is connected neither to the Flight Management System (FMS) nor to the autopilot. Therefore it is designed as an autonomous and independent device that will advise the pilot in the event of an impending collision even if those two systems fail.

A.2.2 Surveillance and Tracking

TCAS II Version 7 tracks proximate aircraft by interrogating their Mode A/C or Mode S transponders. Interrogations are issued at one-second intervals in nominal conditions and in five-seconds intervals in dense traffic airspace, due to the risk of transponder overload caused by a high number of TCAS interrogations. The nominal surveillance range for TCAS II Version 7 is 14 nm for Mode A/C targets and 30 nm for Mode S targets [114]. This range can be reduced to 5 nm in high-density traffic situations to avoid transponder overload. For each aircraft within the surveillance range, TCAS II Version 7 elaborates a surveillance report from each reply received from that aircraft. A surveillance report consists of slant range to the target, relative bearing and altitude. The slant range is calculated from the elapsed time between the emission of the interrogation and the reception of the reply. The relative bearing is obtained from the reply by using the top directional TCAS antenna. The altitude information is included in the surveillance reports elaborated from replies received from targets equipped with altitude-coding transponder.

A.2.3 Collision avoidance Logic

The collision avoidance logic controls the issue of alerts to the pilot. As it has been stated above, TCAS II Version 7 provides two types of alerts: Traffic Advisories and Resolution Advisories. When a target triggers a Traffic Advisory it is labelled as intruder. When a target triggers a Resolution Advisory it is labelled as threat. TCAS II Version 7 estimates the imminent relative trajectories of the targets within the surveillance range by combining surveillance reports elaborated from the targets' replies to consecutive TCAS interrogations. Based upon these predicted trajectories, the collision avoidance logic identifies intruders and threats according to established alerting thresholds.

Essentially, the issue of an alert depends on the estimated time-to-go to the instant when the slant range to the target is estimated to reach a minimum, or Closest Point of Approach (CPA). Additionally, the collision avoidance logic takes into account the estimated miss distance between the TCAS II Version 7-equipped aircraft and the target at the CPA. If the target is equipped with an altitude-coding transponder, the collision avoidance logic is able to provide more efficient protection by using altitude data. Thus, the target's altitude and vertical rate at the CPA can be estimated from consecutive surveillance reports, and the TCAS II Version 7 -equipped aircraft can use its own altitude measures to calculate its vertical rate and the target's relative altitude.

The collision avoidance logic identifies intruders and threats by comparing the time-to-go to the CPA with a time threshold tau (τ). Basically, TCAS II Version 7 issues a Resolution Advisory to the pilot when a target aircraft is in a collision course and the time-to-go to the CPA equals tau. Tau is at the core of the collision avoidance logic of TCAS II Version 7 as the main alerting threshold. Based on the concept of a time threshold tau, the logic establishes two virtual protected volumes around the TCAS II Version 7-equipped aircraft for each target aircraft [115]. When a target enters the external protected volume a TA is issued and the target is identified as an intruder. The TA warns the pilot of the presence of the intruder. When a TA has been issued, the pilot must monitor the intruder since a collision avoidance action might have to be taken shortly should the intruder become a threat. When an intruder enters the internal protected volume, it is identified as a threat and a RA is issued. The RA includes a recommended manoeuvre to avoid the impending collision.

The generation of RAs is explained in the next section. The process of the generation of TAs is analogous. However, in the latter greater alerting thresholds are applied since the aim of the TAs is merely to make the pilot aware of potentially threatening traffic.

A.2.4 Generation of Resolution Advisories

In principle, the internal protected volume defined by the collision avoidance logic has the basic shape of a sphere with a radius equal to the norm of the relative speed vector multiplied by the correspondent value of tau. When tracking altitude-reporting targets, altitude and vertical rate data are taken into account and the sphere is vertically truncated. This truncation aims to reduce the number of unnecessary alerts triggered by targets that enter the sphere defined by tau but which are actually in a course that procures safe vertical separation at CPA. The protected volume is also truncated laterally by a function called Miss Distance Filtering (MDF). The MDF aims to avoid unnecessary alerts in encounters where the predicted horizontal range to the target at the CPA is sufficient to prevent a collision. Moreover, the collision avoidance logic incorporates an additional modification of the protected volume to improve protection against targets approaching at low closure rates. This modification is referred to as DMOD (Distance MODification) and extends the protected volume determined by the warning time tau for targets with a closure rate below a certain threshold. The aim of DMOD is to prevent these targets from getting excessively close in range to the TCAS-equipped aircraft without a RA being triggered.

TCAS II Version 7 also incorporates the concept of Sensitivity Level (SL). The SL varies depending on the altitude of the TCAS-equipped aircraft. The time tau and the DMOD are set according to the SL (See Table A.1). Therefore the size of the protected

Altitude	SL	TAU (s)		DMOD (nm)	
		TA	RA	TA	RA
0-1000 ft	2	20	no RA	0.30	no RA
1000-2350 ft	3	25	15	0.33	0.20
2350 ft-FL050	4	30	20	0.48	0.35
FL050-FL100	5	40	25	0.75	0.55
FL100-FL200	6	45	30	1.00	0.80
>FL200	7	48	35	1.30	1.10

Table A.1: TCAS II Version 7 alert thresholds related to altitude (ACAS Brochure [114]).

volumes depends directly on the SL and consequently on the TCAS II Version 7-equipped aircraft altitude. In general smaller volumes are defined at lower altitudes, where higher traffic densities are expected. This is due to the fact that in high traffic density airspace, the issue of RAs should be restricted to encounters with a high likelihood of resulting in a collision in order to avoid unnecessary traffic disruptions.

A schematic representation of the virtual protected volumes generated by TCAS II Version 7 between FL050 and FL100 is depicted in Figure A.2.

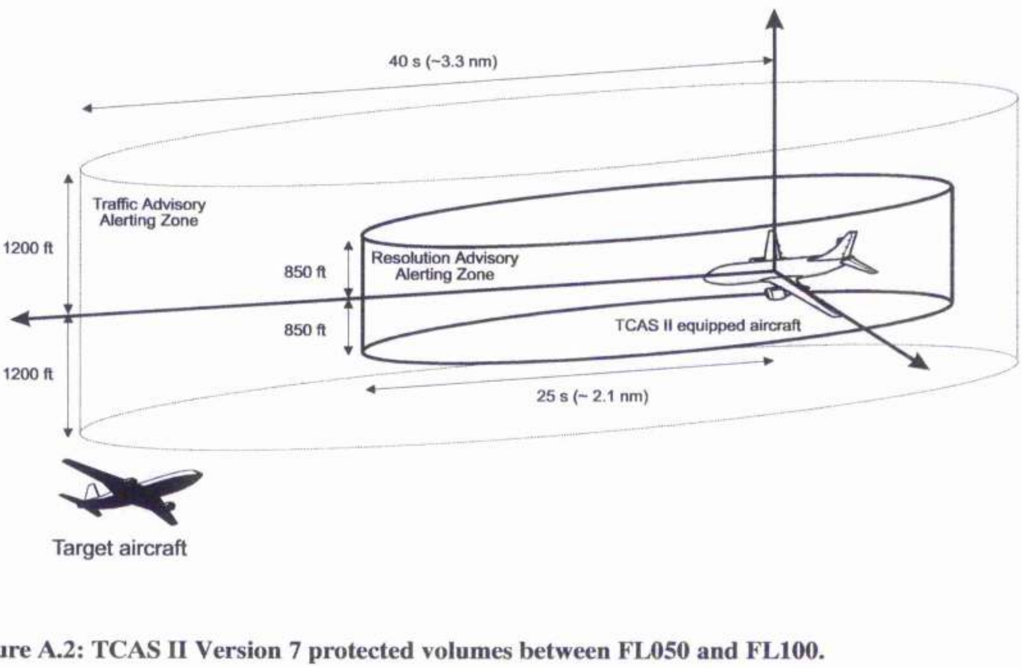


Figure A.2: TCAS II Version 7 protected volumes between FL050 and FL100.

When a target aircraft is declared a threat according to the alerting criteria, the TCAS II Version 7 collision avoidance logic determines the sense (upward or downward) of the avoidance manoeuvre and calculates the least disruptive vertical rate that will achieve safe separation. To establish the vertical avoidance action, the collision avoidance logic considers the predicted courses to the CPA of the threat aircraft and the TCAS II Version 7-equipped aircraft. The threat aircraft is assumed to continue at its current vertical rate and the TCAS II Version 7-equipped aircraft is deemed to start accelerating at 0.25g to the advised vertical rate after a nominal 5 s pilot response delay. RAs are only available for altitude-reporting intruders. The desired vertical safe separation at the CPA after a RA has been implemented varies from 300 ft to 700 ft depending on the TCAS II Version 7-equipped aircraft altitude. Larger values are used at higher altitudes to take into account the increase in barometric altimeter error.

After a RA has been issued to the pilot, TCAS II Version 7 monitors the encounter searching for errors in the trajectory prediction and unexpected manoeuvres of the threat aircraft. As a result of this monitoring, RAs can be modified during the course of an encounter. Thus, the intensity of the recommended manoeuvre is liable to be increased or reduced and, in the event of the threat executing an adverse manoeuvre, the sense of the RA can be reversed. The different Resolution Advisories that can be issued by TCAS II Version 7 are depicted in Table A.2.

In encounters involving two TCAS II Version 7-equipped aircraft, the RAs are co-ordinated. This co-ordination is accomplished by using the Mode S data-link. Essentially, when the aircraft that first detects the impending collision computes its appropriate RA, it transmits information about the chosen RA to the threat via Mode S. Then, before selecting its RA the threat aircraft checks whether it has received any information regarding the other aircraft's imminent RA and, if so, it chooses a RA in the opposite sense.

Furthermore, TCAS II Version 7 is capable of dealing with multi-threat encounters. The collision avoidance logic provides resolutions to these encounters either through a single RA, which will accomplish safe separation from all the threats, or by a RA consisting of a combination of compatible vertical speed restrictions.

Upward sense		Downward sense	
Required vertical rate V (fpm)	**Advisory**	**Required vertical rate V (fpm)**	**Advisory**
+2500	Increase Climb	-2500	Increase Descend
+1500	Climb	-1500	Descend
+1500	Reversal Climb	-1500	Reversal Descend
+1500 fpm	Crossing Climb	-1500	Crossing Descend
+4400>V>+1500	Maintain Climb	-4400<V<-1500	Maintain Descend
V>0	Don't Descend	V<0	Don't Climb
V>-500	Don't Descend>500	V<+500	Don't Climb >500
V>-1000	Don't Descend>1000	V<+1000	Don't Climb >1000
V>-2000	Don't Descend>2000	V<+2000	Don't Climb >2000

Table A.2: Resolution Advisories in TCAS II Version 7 (ACAS Brochure [114]).

A.2.5 Cockpit presentation

The TCAS cockpit presentation comprises information on proximate traffic and aural messages and visual instructions regarding Traffic and Resolution Advisories. A traffic information display indicates the relative horizontal and vertical positions of proximate targets. This traffic information display can be a dedicated TCAS display or can be integrated in the Navigation Display within the Electronic Flight Instrument System (EFIS). The display of the traffic information aims to assist the pilot in the visual acquisition of possible threats. The targets are shown to the pilot using standard symbols according to their ACAS status [114]:

- A hollow blue or white diamond represents non-conflicting surrounding traffic.
- A solid amber circle represents targets that trigger a Traffic Advisory (intruders).
- A solid red square represents targets that trigger a Resolution Advisory (threats).

In addition to the traffic information provided to the pilot in the traffic information display, Traffic and Resolution Advisories are aurally and visually evidenced to the pilot in different forms that vary depending on the cockpit technology. Resolution Advisories may be displayed to the pilot on the artificial horizon, on the vertical speed indicator or on the actual traffic information display.

Appendix B

Candidate data-link technologies for ADS-B

B.1 Introduction

Automatic Dependent Surveillance-Broadcast (ADS-B) is an air-to-air surveillance application currently under development that will enable aircraft to broadcast on-board information to proximate aircraft via data-link. ADS-B is "automatic" because the transmissions occur periodically, requiring no external stimulus to elicit them. ADS-B is "dependent" because the information transmitted is dependent on and derived from the aircraft's on-board sensors. The "surveillance" aspect refers to the primary information transmitted, which is related to the position, velocity, identity and intent of the aircraft. "Broadcast" indicates that the information is continuously broadcast to all proximate users, and is not addressed to a specific receiver. Thus, the aircraft originating the transmissions has no knowledge of which systems are receiving them.

Presently, three different data links are being investigated to support the broadcasting of ADS-B information:

- **1090 MHz Extended Squitter**, which is an extension of the Mode S technology widely used in Secondary Surveillance Radar and TCAS.

- **VDL Mode 4** (Very High Frequency (VHF) Data Link Mode 4), which operates in the VHF frequency range and uses a Self-organising Time Division Multiple Access (STDMA) protocol.

- **UAT** (Universal Access Transceiver), which operates in the 960 MHz band and combines synchronised and random access protocols.

The US Radio Technical Commission for Aeronautics (RTCA) and the European Organisation for Civil Aviation Equipment (EUROCAE) are collaborating in the development of standards for ADS-B applications with different data-link technologies. The US Federal Aviation Administration (FAA) is carrying out ADS-B operational trials using UAT As part of its Capstone initiative [116]. In Europe, the North European ADS-B Network Update Programme (NUP) [117] focuses on the operational introduction of ADS-B with VDL Mode 4. Eurocontrol is working towards the implementation of ADS-B through its ADS programme [118]. Initial findings of this programme suggest that an efficient and reliable ADS-B application will need to incorporate more than one of the proposed data-link technologies.

This appendix provides a brief overview of the main technical characteristics of the three data-link technologies introduced above. Except where specified otherwise, the information in this appendix is excerpted from the Technical Link Assessment Report [119]. This report, commissioned by the FAA's Safe Flight 21 Steering Committee and the Eurocontrol ADS Programme, describes and evaluates the three candidate data-links.

B.2 1090 MHz Mode S Extended Squitter

The 1090 MHz Extended Squitter data-link technology was developed as an extension to the Mode S squitter used in TCAS. Current Mode S transponders periodically broadcast, at a rate of once per second, an unsolicited transmission called *short* (or *acquisition*) *squitter*, which contains 56 bits of information including Mode S control information, the aircraft 24 bit address and parity bits to ensure high integrity decoding.

The short squitter enables TCAS-equipped aircraft to acquire Mode S transponder-equipped aircraft and to carry out surveillance on them using Mode S selective interrogations based on their 24-bit addresses.

The technology of Extended Squitter basically consists of encoding additional information into the short squitters. This results in longer squitter messages, called *extended squitters,* which are periodically broadcast by the Mode S transponder without the need for an external stimulus. The transmission times of the extended squitters are randomised to facilitate multiple access to the 1090 MHz channel.

B.2.1 Waveform

The frequency of the data-link carrier is 1090 ± 1 MHz. The Pulse Position Modulation (PPM) technique is used for data encoding. According to this technique, each bit is assigned a period of time. For each bit period, a pulse is transmitted either in the first half of the period, which indicates a 1, or the second half of the period, which indicates a 0. The data transmission rate is 1Megabits/sec.

B.2.2 Extended squitters and reports

Every transmitted extended squitter includes a 4-pulse preamble that indicates the beginning of the message and enables the receiver to locate and decode the data in it. The extended squitters contain 112 bits, 24 of which are parity bits that can be used for error detection and correction. The term *report* is used to refer to a block of information generated as an output by ADS-B for use as an input to any application. The information in a report is distributed among different extended squitters in the following manner. Position is broadcast in a *position message* transmitted at a rate of 2 per second. Velocity is broadcast in a *velocity message,* which is also transmitted at a rate of 2 per second. The times of applicability of the position and velocity are not transmitted. They are deemed to be the corresponding times of transmission. Additional messages used to generate a report are *aircraft identity messages*, transmitted once per 2.5 seconds, *intent messages* and *status messages*, which are both transmitted once per 1.7 seconds.

A technique for data compression has been developed to encode information efficiently in the bits available in the extended squitters. The technique is called Compact Position Reporting (CPR). The application of the CPR technique in the encoding of position information results in several higher-order bits, which are normally constant for long periods of time, not being transmitted in every message. For example, consider a direct binary representation of latitude in which one bit designates whether the aircraft is in the Northern or the Southern Hemisphere. This bit would remain constant for long periods of time and transmitting it repeatedly in every position message would be inefficient. Using CPR, this bit would not be transmitted. In fact, CPR compresses a 23-bit latitude into a 17-bit one.

As a consequence of several higher-order bits not being transmitted, more than one location on the globe may result in the same encoded message. To avoid ambiguity in the decoding of the aircraft position, the CPR technique uses two encoding formats, called *even-format* and *odd-format*. Each of them is used fifty per cent of the time. Upon reception of messages encoded in different formats within a short period of time (approximately 10 seconds), the receiving system can unambiguously determine the position of the aircraft. The multiple positions corresponding to the messages in the even-format, which are spaced by at least 360 nm, and those corresponding to the messages in the odd-format, which are spaced similarly, coincide only at one point on the globe. Once this process has been carried out, each subsequent single message reception is sufficient to determine the aircraft position unambiguously.

B.2.3 Random time multiple access technique

Whereas the messages are nominally transmitted periodically according to pre-established rates, the actual transmission times are randomly shifted to allow for multiple aircraft to transmit information. Specifically, a timing jitter uniformly distributed over ± 100 ms is applied to each transmission.

B.2.4 Power parameters

The transmitted power levels for extended squitter signals are the same as the existing standards for Mode S transponders: +51 to +57 dBm at the antenna. The receiver sensitivity is characterised by the Minimum Triggering Level (MTL), defined as the power level of a received signal for which correct reception is 90 percent reliable in the absence of interference. It ranges from -74 dBm to -84 dBm, depending on the equipment class.

B.3 VDL Mode 4

VDL Mode 4 is a VHF digital data-link originally developed in Sweden in the late 1980's. It uses two main transmission channels called Global Signalling Channels (GSCs). Access to the data-link is time-multiplexed.

B.3.1 Data-link access

VDL Mode 4 uses a Time Division Multiplex Access (TDMA) structure to enable multiple aircraft to access the data-link and transmit information without interfering with one another. The TDMA structure divides the communication channel into sequential time slots, each of which can be used by an aircraft to transmit information. The slots are grouped into 60 seconds time *superframes* containing 4500 slots. VDL Mode 4 requires a time reference to mark the start of the slots so that aircraft can access them. The time reference for VDL Mode 4 is Universal Time Co-ordinated (UTC), which may be provided by GNSS. Several slot reservation schemes may be used to assign slots to users. The *periodic broadcast protocol* [120] is the most relevant slot reservation scheme for ADS-B applications.

According to the periodic broadcast protocol, which is described in Figure B.1, each aircraft periodically transmits a reservation burst to reserve a slot in each of several subsequent superframes. The burst, which is transmitted during a particular slot (denoted as the "current slot" in Figure B.2), includes the aircraft ID, position information, the *time-out counter* and the *off-set parameter*. The time-out counter indicates in how many subsequent superframes the aircraft reserves the slot that occupies the same position as its "current slot". The off-set parameter indicates the slot reserved by the aircraft in the superframe after the last one in which the slot in the position of the "current slot" is reserved. The aircraft receiving the bursts build a *reservation table*, which contains the slots reserved by the aircraft in its vicinity. When a new user accesses the data-link, it randomly selects a slot among those that are not reserved.

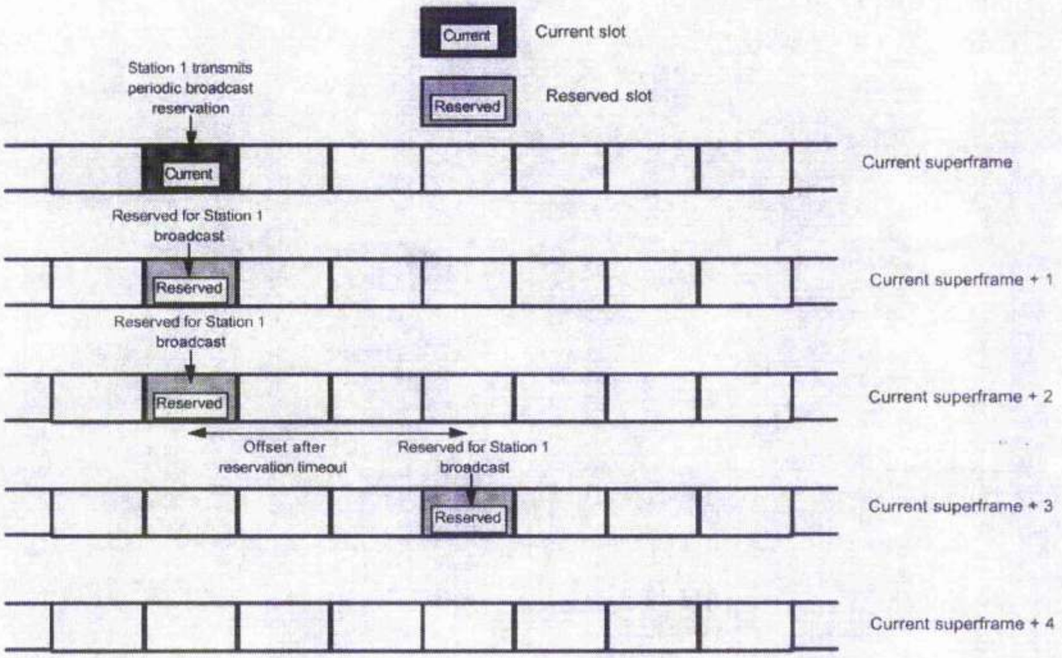


Figure B.1. Periodic broadcast protocol [120]

B.3.2 Waveform, data encoding and messages

VDL Mode 4 operates on 25 kHz-spaced channels in the aeronautical VHF band, which encompasses the frequencies between 108 MHz and 137 MHz. For ADS-B applications, two channels are used (the GSCs).

The modulation technique used in VDL Mode 4 is Gaussian Filtered Frequency Shift Keying (GFSK), which is a continuous-phase, frequency shift keying technique that uses two tones and a Gaussian pulse shape filter. The rate of data transmission is 19,200 bits/sec.

The length of the slots is 256 bits. Position information occupies one slot. More complex information may be transmitted over several slots.

B.3.3 Power parameters

The transmitted power levels for VDL Mode 4 signals at the antenna are between +34 dBm and +47 dBm. The MTL of the receiver is -103 dBm.

B.4 UAT

The UAT was developed at the Mitre Corporation (US) in the mid-nineties. It was specifically designed for data broadcast applications, with simplicity and robustness as paramount design criteria. UTC uses a single frequency in the L-band (960-1215 MHz). Access to the data-link is time-multiplexed.

B.4.1 Data-link access

UAT divides the communication channel into sequential time units called *frames*, which are one second long and begin at the start of each UTC second. Each frame is in turn divided into two segments: one allocated to ground transmissions and the other reserved for ADS-B messages. Each segment is further subdivided into message start opportunities (MSOs) spaced 250 μs apart, which results in a total of 4000 MSOs per frame. The MSO is the smallest time increment used for scheduling ground message or ADS-B message transmissions. The structure of the frame is depicted schematically in Figure B.2.

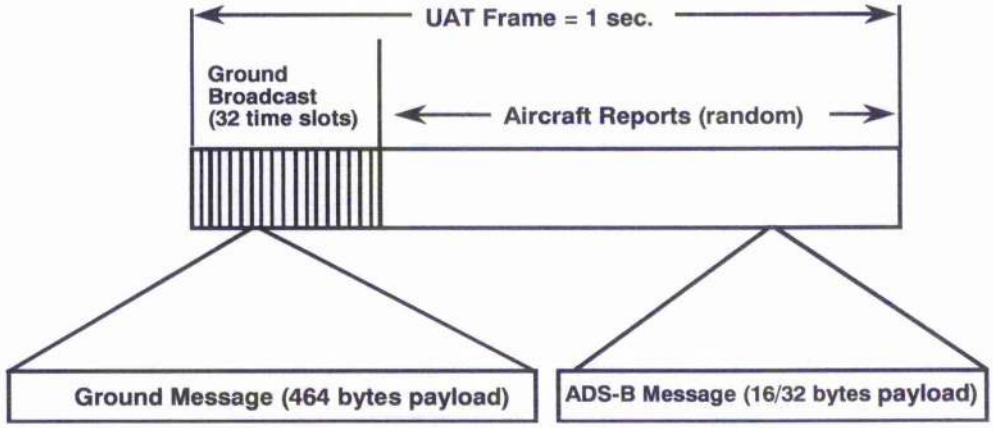


Figure B.2. UAT frame [119]

The ground broadcast segment consists of 752 MSOs, covering a total of 188 ms. These MSOs are grouped into 32 transmission slots of 5.5 ms length, each of them assigned to a different ground station. The ADS-B segment consists of 3248 MSOs. While ground stations are each assigned a fixed transmission slot, the aircraft start transmitting ADS-B messages at randomly selected MSOs from among the first 3200 in the ADS-B segment. Each aircraft transmits one ADS-B message every second. Although the messages are longer than the time between two MSOs, the random selection of the transmission starting time prevents interference among ADS-B transmissions from multiple aircraft.

B.4.2 Waveform, data encoding and messages

UAT operates on a single channel. A carrier frequency of 966 MHz ± 1 MHz has been chosen for most of the tests conducted with UAT so far.

The modulation technique used to encode information in the UAT data-link is Continuous Phase Frequency Shift Keying (CPFSK). A binary 1 is indicated by a $+\frac{\Delta f}{2}$ shift from the nominal carrier frequency f and a binary 0 is indicated by a $-\frac{\Delta f}{2}$ shift. The value of Δf depends on the modulation index and the data transmission rate. The modulation index is 0.6 and the data transmission rate is 1.041677 Megabits/second, which result in Δf = 625 kHz.

The messages may contain either 128 or 256 bits of *payload*, which is the actual ADS-B information encoded in the message. The payload includes the aircraft address, position and velocity. Information describing aircraft intent or airborne meteorological observations could also be transmitted as part of the payload. In addition to the payload, the messages incorporate 124 bits used for synchronisation and error correction.

B.4.3 Power parameters

The transmitted power levels for UAT signals at the antenna are between +44 dBm and +52 dBm. The MTL of the receiver is -92 dBm.

B.5 Summary of the main characteristics of the three data-link technologies

The main characteristics of the three data-link technologies reviewed in this appendix are listed in Table B.1, which is shown in the next page.

	1090 MHz Extended Squitter	VDL Mode 4	UAT
Frequency Band	1090 MHz	108-136.975 MHz	960-1215 MHz
RF Channels	One channel	2 (25KHz) GSCs, plus up to 2 local	One channel
Multiple Access Technique	Random access	Self-organising TDMA	Fixed slots (ground) and random access (aircraft)
Bit Rate	1 Megabits/sec	19,200 bits/sec/channel	1.041667 Megabits/sec
Modulation	PPM	GFSK	CPFSK
Message Length	112 bits	256 bits (slot)	252 bits (short) 380 bits (long)
PVT Segmentation?	Yes: Velocity in separate message	No: PVT in one message	No: PVT in one message
Transmitter Power (at Antenna)	+51 dBm to +57 dBm	+34 dBm to +47 dBm	+44 dBm to +52 dBm
Receiver MTL (90%) (at Antenna)	-84 dBm to -74 dBm	-103 dBm	-92 dBm
Polarisation	Vertical	Vertical	Vertical
Transmission Rate for PVT	Position: 2 Hz Velocity: 2 Hz	Every 10 s en route, 5 s terminal, 1.5 s with local channels	1 Hz
Transmission Rate for intent/ ID	2.2 Hz	Each TCP once every 2.5 min. Flight ID once every 5 minutes	Within same message as PVT

Acronyms and abbreviations:

CPFSK Continuous Phase Frequency Shift Keying
GFSK Gaussian Frequency Shift Keying
GSCs Global Signalling Channels
ID Identification
MTL Minimum Triggering Level
PPM Pulse Position Modulation
PVT Position, Velocity and Time information
RF Radio Frequency
TCP Trajectory Change Point
TDMA Time Division Multiplex Access

Table B.1. Main characteristics of the three data-link technologies

List of acronyms and abbreviations

AAO	Autonomous Aircraft Operations
ACAS	Airborne Collision Avoidance System
ADS	Automatic Dependent Surveillance
ADS-B	Automatic Dependent Surveillance-Broadcast
AI	Artificial Intelligence
ASAS	Airborne Separation Assurance System
ATA	Air Transport Association
ATC	Air Traffic Control
ATFM	Air Traffic Flow Management
ATM	Air Traffic Management
ATN	Aeronautical Telecommunications Network
BCAS	Beacon Collision Avoidance System
CDM	Collaborative Decision Making
CDTI	Cockpit Display of Traffic Information
CFIT	Controlled Flight Into Terrain
CFMU	Central Flow Management Unit
CNS	Communications, Navigation and Surveillance
CPA	Closest Point of Approach
CPFSK	Continuous Phase Frequency Shift Keying
CPR	Compact Position Reporting
CPU	Central Processing Unit
DAI	Distributed Artificial Intelligence
dBm	decibels relative to 1 milliwatt
DMOD	Distance MODification
EACAC	Evolutionary Air-ground Co-operative ATM Concept
ECAC	European Civil Aviation Conference
EFIS	Electronic Flight Instrument System

EFR	Extended Flight Rules
EROS	Eliminate Range-zerO System
EUROCAE	European Organisation for Civil Aviation Equipment
FAA	Federal Aviation Administration
FACES	Free Flight Autonomous Embarked Solver
FANS	Future Air Navigation System
FAST	Full Autonomous Separation Transfer
FFAS	Free Flight Airspace
FFP1	Free Flight Phase 1
FL	Flight Level
FMS	Flight Management System
fpm	feet per minute
FREER	Free-route Experimental Encounter Resolution
ft	feet
g	Acceleration of gravity (9.8 ms^{-2})
GEARS	Generic En-route Algorithmic Resolution Service
GFSK	Gaussian Frequency Shift Keying
GNSS	Global Navigation Satellite System
GSC	Global Signalling Channels
Hz	Hertz
ICAO	International Civil Aviation Organisation
ID	Identification
IERE	Institution of Electrical and Radio Engineers
kHz	kilohertz
kt	knots (nautical miles/hour)
MAS	Managed Airspace
MB	Mega Bytes
MDF	Miss Distance Filtering

MHz	Megahertz
min	minutes
M-MRMH	Modified Multi-start Random Mutation Hillclimbing
ms	milliseconds
MSO	Message Start Opportunity
MTL	Minimum Triggering Level
NAS	National Airspace System
NLR	National Aerospace Laboratory (The Netherlands)
nm	nautical miles
NUP	Network Update Programme
OCD	Operational Concept Document
PASAS	Predictive ASAS
PC	Personal Computer
PDF	Probability Density Function
pFAST	passive Final Approach Spacing Tool
PPM	Pulse Position Modulation
PVT	Position, Velocity and Time information
RA	Resolution Advisory
RAM	Random Access Memory
RF	Radio Frequency
RGCS	Review of the General Concept of Separation
RMM	Recursive Modelling Method
RNAV	Area Navigation
RTCA	Radio Technical Commission for Aeronautics
s	seconds
SICAS	Surveillance Radar Improvements and Collision Avoidance Systems
SL	Sensitivity Level
SMA	Surface Movement Advisor
SQP	Sequential Quadratic Programming

SSR	Secondary Surveillance Radar
TA	Traffic Advisory
TCAS	Traffic Collision Avoidance System
TCP	Trajectory Change Point
TDMS	Time Division Multiplex Access
TIS-B	Traffic Information service-Broadcast
TLS	Target Level of Safety
TMA	Terminal Manoeuvring Area
TSA	Traffic Situational Awareness
UAT	Universal Access Transceiver
UMAS	Unmanaged Airspace
URET	User Request Evaluation Tool
US	United States of America
UTC	Universal Time Co-ordinated
VDL	Very High Frequency Data-link
VHF	Very High Frequency

References

1. Kayton, M., Fried, W. R., *Avionics navigation systems, 2nd edition*. 1997, New York: John Wiley & Sons, Inc.

2. *Executive Summary of the Global Air Navigation Plan for CNS/ATM Systems*. in *ICAO World-wide CNS/ATM Systems Implementation Conference*. 1998. Rio de Janeiro.

3. *Outlook for Air Transport to the Year 2005*. 1997, Circular 270. ICAO.

4. Pierce, J.I., *Air Traffic Management in Developing Regions*. in *48th Annual International Air Safety Seminar*. 1995. Seattle, Washington. USA.

5. *Final Report of RTCA Task Force 3; Free Flight Implementation*. 1995, Radio Technical Commission for Aeronautics.

6. *FAA Free Flight Phase 1 website.*, http://ffp1.faa.gov.

7. *FAA Safe Flight 21 website.*, http://www.faa.gov/safeflight21/.

8. *European Organisation for the Safety of Air Navigation (EUROCONTROL) website.*, http://www.eurocontrol.be/.

9. *European Civil Aviation Conference (ECAC) website.*, http://www.ecac-ceac.org/.

10. *EATM Strategy for 2000+*. 2000, European Organisation for the Safety of Air Navigation (EUROCONTROL).

11. *European Air Traffic Management System Operational Concept Document (OCD). Edition 1.1*. 1999, European Organisation for the Safety of Air Navigation (EUROCONTROL).

12. *Annex 2 to the Convention on International Civil Aviation. Rules of the air.* 1999, ICAO.

13. SICASP/WG2-489, *Airborne Separation Assurance System - The ASAS concept*. 1995, ICAO: Sydney.

14. SICASP/6/WP44G2, *The ASAS concept*. 1997, Report on Agenda Item 6. ICAO: Montreal.

15. SICASP/WG2/WP2, *Operational and technical considerations on ASAS applications*. 1999, ICAO.

16. *Principles of Operations for the Use of ASAS (PO-ASAS).* FAA-Eurocontrol R&D Committee. Action Plan 1 Team. 2002. http://human-factors.arc.nasa.gov/ihi/documents/PO-ASAS.pdf.

17. SICASP/WG2-WP2/724, *ASAS Definitions.* 1998, ICAO: Brussels.

18. *WP5: Assessment of emerging technologies: the specific case of ADS-B/ASAS.* 1998, Project EMERALD Technical Report. European Commission.

19. Abeloos, A., Mulder, M., Paasen, R. van, Hoffman, E., *Potential co-operation between the TCAS and the ASAS.* 2000, EUROCONTROL Experimental Centre.

20. ADSP/WGA/WP18, *Automatic Dependent Surveillance-Broadcast (ADS-B). Concept Paper.* 1999, ICAO.

21. *Minimum Aviation System Performance Standards for Automatic Dependent Surveillance-Broadcast (ADS-B MASPS).* 1998, RTCA DO-242. Radio Technical Commission for Aeronautics.

22. *Review of the General Concept of Separation Panel (RGCSP). Tenth Meeting.* 2000, ICAO: Montreal.

23. Duong, V.N., Hoffman, E., Nicolaon, J. -P. *Conflict Resolution Advisory Service in Autonomous Aircraft Operations.* in *16th IEEE Digital Avionics System.* 1997. Irvine, California, USA.

24. Duong, V.N., Hoffman, E., Nicolaon, J. -P. *Initial Results of Investigation into Autonomous Aircraft Concept (FREER-1).* in *1st USA/Europe ATM R&D Seminar.* 1997. Saclay, France.

25. McGibbon, F., Goodchild, C. *Autonomous aircraft operations to managed airspace transfer management tool (T-MAT).* in *20th Digital Avionics Systems Conference.* 2001. Daytona Beach (USA): IEEE, Piscataway, NJ, USA.

26. Rawlings, R., Adam, V., Ingle, G. *Experimental Flight Management System.* in *56th Symposium on Machine Intelligence in Air Traffic Management.* 1993. Berlin, Germany: Advisory Group for Aerospace Research and Development (AGARD), Guidance and Control Panel.

27. Simpson, R.W. *Structuring Criteria for Automated Separation Assurance.* in *2nd USA/Europe ATM R&D Seminar.* 1998. Orlando, USA.

28. Warren, A., *Medium Term Conflict Detection for Free Routing: Operational Concepts and Requirements Analysis.* 1997, Boeing Commercial ACFT Group: Seattle.

29. Krozel, J., Peters, M. *Strategic conflict Detection and Resolution for Free Flight.* in *36th Conference on Decision and Control.* 1997. San Diego, California. USA.

30. *Freer Flight website.*, http://www.eurocontrol.fr/projects/freer/.

31. Hoffman, E., Zeghal, K., Cloerec, C., Grimaud, I., Nicolaon, J.-P., *Operational concepts for limited delegation of separation assurance to the cockpit.* 1999, AIAA 99-3990.

32. Hoffman, E., Zeghal, K., Cloerec, C., Grimaud, I., Nicolaon, J.-P. *Is limited delegation of separation assurance promising?* in *3rd USA/Europe Air Traffic Management R&D Seminar.* 2000. Napoli (Italy).

33. Duong, V.N., Zeghal, K. *Conflict Resolution Advisory for Autonomous Airborne Separation in Low-density Airspace.* in *36th IEE Conference on Decision and Control.* 1997. San Diego, CA. USA.

34. Hoffman, E., Grandperret, S. *Pilot-in-the-loop Evaluation of Cockpit Assistance for Autonomous Operations.* in *SAE/AIAA World Aviation Congress.* 2000. San Diego, USA.

35. Duong, V.N., Hoffman, E., Flochic, L., Nicolaon, J.-P., Bossu, A., *Extended Flight Rules (EFR) to apply to the resolution of encounters in autonomous airborne separation.* 1996, EUROCONTROL Experimental Centre.

36. Irvine, R., *The GEARS Conflict Resolution Algorithm.* 1997, EUROCONTROL Experimental Centre.

37. Hoekstra, J.M., van Gent, R.N.H.W., Ruigrok, R.C.J. *Conceptual Design of Free Flight with Airborne Separation Assurance.* in *AIAA Guidance, Navigation and Control Conference.* 1998. Boston, USA.

38. Hoekstra, J.M., *Designing for safety: Free Flight Air Traffic Management Concept.* 2001, PhD Thesis. Delft University of Technology: Delft (The Netherlands).

39. van Gent, R.N.H.W., Hoekstra, J., Ruigrok, R.C.J. *Free Flight with Airborne Separation Assurance.* in *Human Computer Interaction Conference.* 1998. Montreal, Canada.

40. Eby, M.S., *A Self-Organizational Approach for Resolving Air Traffic Conflicts.* The Lincoln Laboratory Journal, Vol 7, No 2. 1994.

41. Alliot, J.M., Durand, N., Granger, G. *FACES: a Free Flight Autonomous and Co-ordinated Embarked Solver*. in *2nd USA/Europe Air Traffic Management R&D Seminar*. 1998. Oralndo (USA).

42. Barr, A., Feigenbaum, E., ed. *The Handbook of Artificial Intelligence, Volumen 1*. 1981, Pitman Books Limited: London.

43. Hoekstra, J.M., van Gent, R.N.H.W., Ruigrok, R.C.J. *Free Flight in a Crowded Airspace?* in *3rd USA/Europe ATM R&D Seminar*. 2000. Napoli (Italy).

44. Schild, R., *Rule Optimization for Airborne Aircraft Separation*. 1998, PhD Thesis. Technical University of Vienna: Vienna (Austria).

45. Schild, R., Kuchar, J.K. *Operational efficiency of maneuver coordination rules for an airborne separation assurance system*. in *3rd USA/Europe Air Traffic Management R&D Seminar*. 2000. Napoli (Italy).

46. Goldberg, D.E., *Genetic algorithms in search, optimization and machine learning*. 1989, Reading, MA: Addison-Wesley.

47. Tomlin, C., Pappas, G. J., Sastry, S. *Noncooperative Conflict Resolution*. in *36th Conference on Decision and Control*. 1997. San Diego, California. USA.

48. Kosecka, J., Tomlin, C., Pappas, G. J., Sastry, S. *Generation of Conflict Resolution Manoeuvers*. in *IEEE International Conference on Intelligent Robots and Systems*. 1997.

49. Tomlin, C., Pappas, G. J., Sastry, S., *Conflict resolution for Air Traffic Management: A study in Multiagent Hybrid Systems*. IEEE Transactions on Automatic Control, Vol 43, No 4. 1998: p. 509-521.

50. Tomlin, C., *Hybrid Control of Air Traffic Management Systems*. 1998, PhD thesis. University of California at Berkeley.

51. Tomlin, C., Mitchell, I., Ghosh, R., *Safety Verification of Conflict Resolution Maneuvers*. IEEE Transactions on Intelligent Transportation Systems, Vol 2, No 2. 2001.

52. Smith, K., Briggs, A., Knecht, W., Hancock, P. *Success and failure at self-separation in simulated Free Flight*. in *41st Annual Meeting of the Human Factors and Ergonomics Society*. 1997. Alburquerque (USA).

53. Cashion, P., Mackintosh, M.-A., McGann, A., Lozito, S. *A study of commercial flight crew self-separation*. in *16th AIAA/IEEE Digital Avionics Systems Conference*. 1997. Irvine (USA).

54. Yang, L.C., Kuchar, J. K., *Prototype Conflict Alerting System for Free Flight.* Journal of Guidance, Control, and Dynamics, Vol 20, No 4. 1997: p. 768-773.

55. Zeghal, K. *A Review of Different Approaches Based on Force Fields for Airborne Conflict Resolution.* in *AIAA Guidance, Navigation and Control Conference.* 1998. Boston, USA.

56. Zeghal, K. *A Comparison of Different Approaches Based on Force Fields for Coordination among Multiple Mobiles.* in *IEEE International Conference on Intelligent Robotic Systems (IROS).* 1998. Victoria (Canada).

57. Hu, J., Lygeros, J., Prandini, M., Sastry, S., *Aircraft Conflict Prediction and Resolution using Brownian Motion.* 1998, University of California, Berkeley. NASA grant NAG 2-1039.

58. Menon, P.K., Sweriduk, G. D., *Optimal Strategies for Free-Flight Air Traffic Conflict Resolution.* Journal of Guidance, Control, and Dynamics, Vol 22, No 2. 1999: p. 202-211.

59. Russell, S., Norvig, P., *Artificial Intelligence: a modern approach.* Prentice Hall Series in Artificial Intelligence. 1995, New Jersey: Prentice-Hall,Inc.

60. Wooldridge, M., *The Logical Modelling of Computational Multi-Agent Systems.* 1992, PhD thesis, UMIST: Manchester.

61. Singh, M.P., Rao, A. S., Georgeff, M. P., *Formal Methods in DAI: Logic-Based Representation and Reasoning,* in *Multi-agent Systems: A Modern Approach to Distributed Artificial Intelligence,* G. Weiss, Editor. 1999, MIT Press. p. 331, 376.

62. Wooldridge, M., Jennings, N., editors, *Intelligent Agents.* 1995, New York: Springer-Verlag.

63. Kaiser, M., Klingspor, V., Friedrich, H. *Human-agent Interaction and Machine Learning.* in *9th European Conference on Machine Learning.* 1997. Prague, Czech Republic: Springer.

64. Wooldridge, M., Jennings, N., *Intelligent Agents: Theory and Practice.* Knowledge Engeneering Review, Vol 10, No 2. 1995.

65. Pollock, J.L., *Planning Agents,* in *Foundations of Rational Agency,* A. Rao, Wooldridge, M., Editor. 1999, Kluwer Academic Publishers.

66. Jennings, N., *Cooperation in Industrial Multi-Agent Systems.* 1994, Singapore: World Scientific Publishing Co. Pte. Ltd.

67. Lesser, V.R., *Cooperative Multiagent Systems: A Personal View of the State of the Art.* IEEE Transactions on Knowledge and Data Engineering, Vol 11, No 1. 1999.

68. Doran, J.E., Franklin, S., Jennings, N. R., Norman, T. J., *On Cooperation in Multi-Agent Systems.* 1996, First UK Workshop on Foundations of Multi-Agent Systems: University of Warwick.

69. Uny Cao, Y., Fukunaga, A. S., Kahng, A. B., *Cooperative Mobile Robotics: Antecedents and Directions.* Autonomous Robots, No 4. 1997: p. 1-23.

70. Werger, B.B., *Cooperation without deliberation: A minimal behavior-based approach to multi-robot teams.* Artificial Intelligence, No 110. 1999.

71. Evans, K.S., Unsal, C., Bay, J. S., *A Reactive Coordination Scheme for a Many-Robot System.* IEEE Transactions on Systems, Man and Cybernetics, Vol 27, No 4. August 1997: p. 598-610.

72. Schoonderwoerd, R., Holland, O., Bruten, J., Rothkrantz, L., *Ants for load balancing in telecommunication networks.* 1996, Hewlett Packard Labs Technical Report.

73. Wagner, I.A., Lindenbaum, M., Bruckstein, A. *Cooperative Covering by Ant-Robots using Evaporating Traces.* in *4th Israeli Symposium on Theory of Computing and Systems.* 1996. Jerusalem (Israel).

74. Beslon, G., Biennier, F., Hirsbrunner, B. *Multi-Robot Path Planning Based on Implicit Cooperation in a Robotic Swarm.* in *2nd International Conference on Autonomous Agents (Agent'98).* 1998. Minneapolis (USA).

75. Alami, R., Ingrand, F., Qutub, S. *A Scheme for Coordinating Multi-robot Planning Activities and Plans Execution.* in *ECAI 98. 13th European Conference on Artificial Intelligence.* 1998: John Wiley & Sons, Ltd.

76. Laird, J.E., Jones, R. M., Nielsen, P. E., *Knowledge-Based Multi-Agent Coordination.* 1998, University of Michigan: Soar/IFOR Project.

77. Goldman, C.V., Rosenschein, J. S. *Emergent coordination through the use of cooperative state-changing rules.* in *12th National Conference on Artificial Intelligence.* 1994. Philadelphia (USA).

78. Gmytrasiewicz, P.J., Durfee, E. H., *Rational Coordination in Multi-Agent Environments.* Autonomous Agents and Multi-Agent Systems, Vol 3, No 4. 2000: p. 319-350.

79. Hughes, G.E., Cresswell, M. J., *A new introduction to modal logic*. 1996, London: Routledge.

80. Grinton, C., Sonenberg, L., Sterling, L. *Exploring agent cooperation: studies with a simple pursuit game*. in *Australian Artificial Intelligence Conference*. 1997.

81. Davis, R., Smith, R. G., *Negotiation as a metaphor for Distributed Problem Solving.* ", Artificial Intelligence, Vol 20. 1983: p. 63-109.

82. Stone, P., Veloso, M., *Task decomposition, dynamic role assignment, and low-bandwidth communication for real-time strategic teamwork*. Artificial Intelligence, Vol 110. 1999: p. 241-273.

83. Tambe, M., Adibi, J., Al-Onaizan, Y., Erdem, A., Kaminka, G. A., Marsella, S. C., Muslea, I., *Building agent teams using an explicit teamwork model and learning*. Artificial Intelligence, Vol 110. 1999: p. 215-239.

84. Barrer, J. *Integrating the Cockpit with Air Traffic Management: the Concept of Path Objects*. in *3rd USA/Europe ATM R&D Seminar*. 2000. Napoli (Italy).

85. Basar, T., Jan Olsder, G., *Dynamic Noncooperative Game Thery*. Mathematics in Science and Engineering, ed. R. Bellman. Vol. 160. 1982, London (UK): Academic Press Inc.

86. Dorn, J., Girsch, M., Skele,G., Slany, W. *Comparison of Iterative Improvement Techniques for Schedule Optimization*. in *13th UK Planning Special Interest Group*. 1994. Glasgow (UK).

87. Dorn, J., *Iterative Improvement Methods for Knowledge-based Scheduling*. AI Communications, Vol 8. No 1. 1995: p. 20-34.

88. Skiena, S.S., *The algorithm design manual*. 1997, New York: TELOS/Springer-Verlag.

89. Slattery, R., Zhao, Y., *Trajectory Synthesis for Air Traffic Automation*. Journal of Guidance, Control and Dynamics, Vol 20, No 2. 1997: p. 232-238.

90. Bos, A., *User Manual for the Base of Aircraft Data (BADA) - Revision 2.6*. 1997, EEC Note No 23/97. EUROCONTROL.

91. *Using MATLAB, Version 5*. 1998, The Mathworks, Inc.: Natick, MA, USA. http://www.mathworks.com.

92. Papoulis, A., *Probability, Random Variables, and Stochastic Processes*. 3rd ed. Electric & Electronic Engineering Series. 1991, Singapore: McGraw-Hill International Editions.

93. Ott, L., Mendenhall, W., *Understanding Statistics, 4th Edition*. 1985, Boston (USA): PWS Publishers.

94. McCarthy, J., *Formalizing Common Sense*, ed. V. Lifschitz. 1990: Ablex.

95. Tuomela, R., *Shared Belief*, in *International Encyclopedia of the Social and Behavioral Sciences*, P. Baltes, Smelser, N., Editor. 2001, Elsevier Science: Oxford (UK).

96. Osyczka, A., *Multicriterion optimization in engineering with FORTRAN programs*. 1984, England: Ellis Horwood series in engineering science.

97. Zitzler, E., *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*. 1999, PhD thesis, Computer Engineering and Networks Laboratory, Swiss Federal Institute of Technology: Zurich.

98. Pilegaard Hansen, M. *Tabu Search for Multiobjective Optimization: MOTS*. in *13th International Conference on Multiple Criteria Decision Making*. 1997. Cape Town, South Africa.

99. Knowles, J.D., *Local-search and Hybrid Evolutionary Algorithms for Pareto Optimization*. 2002, PhD thesis, Department of Computer Science, University of Reading: Reading.

100. Voss, S., *Meta-heuristics: The State of the Art*, in *Local Search for Planning and Scheduling*, A. Nareyek, Editor. Lecture Notes in Artificial Intelligence 2148. 2001, Springer: Berlin.

101. Yagiura, M., Ibaraki, T., *On Metaheuristic Algorithms for Combinatorial Optimization Problems.* Transactions of the Institute of Electronics, Information and Communication Engineers, Vol J83-D-I, No1. 2000: p. 3-25.

102. Blum, C., Roli, A., *Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison*. Technical Report No TR/IRIDIA/2001-13. 2001, IRIDIA (Artificial Intelligence Research Laboratory of the Université Libre de Bruxelles): Brussels.

103. Aarts, E., Lenstra, J. K., ed. *Local Search in Combinatorial Optimisation*. 1997, John Wiley & Sons Ltd: Chichester, UK.

104. Lourenco, H., Martin, O. C., Stutzle, T, *Iterated Local Search*, in *Handbook on Metaheuristics*, F. Glover, Kochenberger, G., Editor. To appear in 2003, Kluwer Academic Publishers: Norwell, USA. Preliminary version in http://www.intellektik.informatik.tu-darmstadt.de/~meta/public.html.

105. Kirkpartick, S., Gelatt, C. D., Vecchi, M. P., *Optimization by simulated annealing*. Science, Vol 220, No 4598. 1983.

106. Glover, F., Laguna, M., *Tabu Search*. 1997, Boston: Kluwer Academic Publishers.

107. Dorigo, M., Di Caro, G., *The Ant Colony Optimization meta-heuristic*, in *New Ideas in Optimization*, D. Corne, Dorigo, M., Glover, F., Editor. 1999, McGraw-Hill: London.

108. Ricker, R., *Traffic Alert/Collision Avoidance System*. 1999, Aeronautics Learning Laboratory for Science, Technology and Research. http: //www.allstar.fiu.edu/aero/TCAS.htm.

109. Morrel, J.S., *The Mathematics of Collision Avoidance in the Air*. Journal of Navigation, No 11, 1958.

110. *Procedures for Air Navigation Services-Rules of the Air and Air Traffic Services.*, Document 4444 of ICAO. PANS-RAC. ICAO.

111. *Procedures for Air Navigation Services-Aircraft Operations.*, Document 8168 of ICAO. PANS-OPS. ICAO.

112. *Annex 10 to the Convention on International Civil Aviation. Aeronautical communications. Volume IV (Surveillance radar and collision avoidance systems)*. 1998, ICAO.

113. *FAR 121.356: Traffic Alert and Collision Avoidance System.*, Federal Aviation Administration Regulations.

114. *ACAS brochure, Version 1.0*. 1999, CENA and Eurocontrol for ACAS Analysis European Project.

115. Ford, R.L., *The protected volume of airspace generated by an Airborne Collision Avoidance System*. Journal of Navigation, Vol 39 1958: p. 139-158.

116. *FAA Alaskan Region Capstone Program.*, http://www.alaska.faa.gov/capstone/.

117. *North European ADS-B Network Update Programme.*, http://www.nup.nu/.

118. *Eurocontrol ADS Website.*, http://www.eurocontrol.int/ads.

119. *Technical Link Assessment Report*. 2001, ADS-B Technical Link Assessment Team (Eurocontrol ADS Programme) http://www.eurocontrol.int/ads/s1.

120. *VDL Mode 4 in CNS/ATM. Master Document*. 2001, NUP Project. http://www.nup.nu/.