Airborne Sense and Alert Collision Warning and Avoidance System

Liwaa H. Alnashi

A Thesis submitted to the Auckland University of Technology in fulfilment of the requirements of the Degree of Masters of Engineering (MEng)

AUT UNIVERSITY

TE WĀNANGA ARONUI O TAMAKI MAKAU RAU

Primary Supervisor: Dr. John Collins

August 2011

CERTIFICATE OF ORIGINALITY

'I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgements), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning'

Liwaa H. Alnashi

August 2011

ACKNOWLEDGEMENT

This research could not have been completed without Dr. John Collins' support and trust. I would like to thank Dr. Collins for his dedication, attention, and his ever welcoming attitude towards me.

I would also like to thank my colleagues at TGR Helicorp, namely Chris Morris, Rikk Higgot, and Ben Jones for their support and care.

Airborne Sense and Alert

Collision Warning and Avoidance System

ABSTRACT

The emergence of unmanned aerial vehicles and the ever increasing performance in terms of speed, flight altitude, and endurance have led the debate into allowing such aircraft to fly co-jointly with civilian aircraft. However, due to the absence of the usual flight regulation channel, not to mention the pilot at the aircraft controls, the need for an automated collision avoidance system has arisen in the past few years.

For that we propose to provide a practical solution to equip UAVs with an autonomous sense and avoid capability and an autonomous collision avoidance system, to enable the UAV to fly in a non-segregated air space safely and meeting the above regulations.

In this research, we have evaluated different types of mechanism to form this collision avoidance system. The most successful of which concluded of path estimation and the calculation of nearest point of approach. During this research, we developed a collision avoidance mechanism that uses vector algorithms and path estimation methods to increase the efficiency of the logic system and decrease the computation time.

In the results of this experiment, we determined that using few well developed manoeuvres would result in better avoidance efficiency and would require limited change to the flight path of the unmanned vehicle and flight parameters.

Manoeuvres such as changing speed or turning provided the best options for avoiding incoming aircraft, while changing altitude was less successful due to the danger of flying into a different flight level (sharing the same altitude levels with other aircraft) and due to the limited climb and decent rate performances of the model unmanned aerial vehicle used.

More complicated scenarios, such as avoiding multiple aircraft would require a slightly different strategy, where the algorithm would be based upon avoiding a flight path of all aircraft at all times, rather than changing velocity or heading to avoid colliding at a certain point in time.

The main outcome of this experiment, was to prove that such algorithms (with limited complex theory behind it) can prove to be a good option for deriving avoidance systems and ensuring flight safety for manned and unmanned aircraft.

Testing was successfully conducted by the student on simple implementation of the Loss of Separation algorithm to verify, test and expand the algorithm as a pre-preparation for code integration in later stages.

In this document, we present how we will implement the “sense and avoid” algorithm and the logical decision making system that would provide the UAV with the ability to re-route its current path to a safer flight course.

CONTENTS

LIST OF FIGURES

LIST OF EQUATIONS

LIST OF TABLES

ACRONYMS & ABBREVIATIONS

ACAS	Airborne Collision Avoidance System
API	Advanced Programming Interface
ASTM	American Society for Testing and Materials
ATC	Air Traffic Controllers/Control
CAA	Civil Aviation Authority
CFR	Code of Federal Regulation
COA	Certificate of Authority
CPA	Closest Point of Approach (corresponds to separation distance “d”)
CPU	Central Processing Unit
DMA	Direct Memory Access
DSP	Digital Signal Processing
EADS	European Aeronautics Defence & Space
EMI	Electro-Magnetic Interference
FAA	Federal Aviation Authority
HALE	High Altitude Long Endurance
ICAO	International Civil Aviation Organization
IFR	Instrument Flight Rules
ITAR	International Trade in Arms Regulations
NASA	National Aeronautics and Space Administration
PCB	Printed Circuit Board
ROT	Rate of Turn

RTCA	Radio Technical Communication for Aeronautics
RVSM	Reduced Vertical Separation Minima
SAR	Synthetic Aperture Radar
SBC	Single Board Computer
SDRAM	Synchronous Dynamic Random Access Memory
TCA	Time to Closest/Nearest Approach
TCAS	Traffic alert and Collision Avoidance System
UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicle
ULP	Ultra Low Power
VFR	Visual Flight Rules
WP	Way Point

CHAPTER 1

INTRODUCTION

The evolving capabilities of unmanned aerial vehicles (UAVs) have captured the attention of the civilian field, potentially creating new opportunities and markets for their multitude capacities. Coupled with the military complex's intention of fielding unmanned combat aerial vehicles in greater numbers, this has created a plethora of military and civilian enthusiasts alike.

Some modern unmanned flying weapons platforms surpasse conventional manned aircraft in terms of travelling long ranges at high altitudes, and loitering time over target. Such performance advantages have encouraged military leaders and civilian visionaries to request that UAVs be allowed access into international flight routes in order to capitalize on their performance and expand their area of operation.

However, limitations in areas such as reliable obstacle detection and collision avoidance systems have restricted the use of UAVs to ground control and human intervention in course planning and control, in addition to the confinement of this type of aircraft to segregated air space

As the technology advances enable autonomous long endurance missions to be conducted using UAVs, the human factor of ground control must be eliminated. Due to the lack of successful and trustworthy mechanisms of collision avoidance, UAVs have been restricted into flying in specific air space cordons by the aviation regulatory authorities, as they are yet to be convinced that the technology is safe enough to venture into civilian or commercial airspace.

The greatest challenge to automated flight systems (especially those applied in 4th generation autopilots and unmanned aerial vehicles) is maintaining a safe separation between aircraft sharing a defined volume of air space, a problem commonly known as "Conflict Detection and Resolution" (CD&R). [1]

The purpose of this research is to provide detailed information on the design methodology and process, software design and testing for the collision avoidance program/algorithms of the ***Sense and Avoid*** project.

This document will include descriptions and discussions concerning chosen hardware and software options that can be used to design and implement a collision avoidance system.

CHAPTER 2

LITERATURE REVIEW

The concept of UAVs sharing the same airspace as civilian airliners is causing a great debate in the aviation industry. Government bodies and aviation authorities oppose allowing this integration of airspace to occur for obvious safety concerns.

In their report the US Department of Transportation clarified that the most daunting scenario would be a mid-air collision with a commercial or personal aircraft. [2]

Depending on the size and the altitude of the UAV platform used for surveillance mission, ground crew distraction may lead to a disasterous accident. However, the recent increases in flight duration, size, flight speeds, and altitude of the newer generations of UAVs would eventually find them included as transport vehicles. These would almost certainly share the same airspace and international flight routes as piloted passenger aircraft.

This has ultimately raised some concerns in both the aviation transport industry and civil aviation authorities; these concerns mainly involve flight safety factors, namely the possibility of mid air collision that may occur due to equipment failure of the automatic pilots of UAVs, or the occasional mishaps that occur due to human error in planning flight routes or management of airspace.

Therefore one of the most enduring projects run by civil aviation bodies over the past three decades has revolved around the development of automated collision avoidance systems for both manned and unmanned aircraft.

The requirement is that collision avoidance systems must be able to sense and avoid other aircraft within the planned operational course in order to provide the same level of safety as a manned aircraft.

2.1 The Need for Collision Avoidance

The dominance of air travel as the most preferred mode of travel for business, tourism and commercial traffic and goods exchange has been one of the most distinguishing features of the later part of the 20^{th} century. This demand spawned multi-national conglomerates that design, build, sell and maintain large aircraft capable of transporting hundreds of passengers across distances of thousands of kilometres. The continuous reduction in air transportation costs, have increased the number of passengers and goods exchanged between countries, and in turn the number of commercial travel businesses and therefore the number of aircraft concurrently flying in the same air sector. [3]

Unfortunately, this increase in the number of aircraft has been coupled with longer flight delays and longer waiting (or transit) periods at airports, resulting in reduced operational efficiencies for the carriers and airports.

To counter this problem, the International Civil Aviation Organization (ICAO) recommended reducing the lateral and vertical separation gaps between the flight-paths of different aircraft from 2000ft to 1000 ft. [3]

This measure is intended to maximise the number of civilian aircraft occupying a given volume at any time. However, this regulation has raised safety concerns within the aviation industry due to the increased probabilities of mid-air collision.

The reduction in separation distances between aircraft would reduce the pilot’s response time considerably, from approximately 45 seconds to below 30 seconds.

In most cases the available response time will be approximately 20 seconds, far too short for a human pilot to identify the danger, estimate the time of collision, calculate which action to take, and physically change the flight path of the aircraft. [4]

To complicate the problem even further, the advent of newer UAV and private aircraft, capable of high-speed high-altitude flight, has ushered the need for more comprehensive legislations to better regulate the airspace. [5]

However, the introduction of automated flight controllers in both civilian aircraft and UAV systems might present a practical solution to the problem. Then the human response factor can be removed from the avoidance process, and the aircraft (airliner, private, or UAV) can autonomously detect, assess and avoid collision situations.

Mid-air collisions are not a new problem, and during the last decade, several solutions presented themselves as useful means to detect collision/close proximity dangers and to assist the pilot in identifying dangerous situations. One of the most commonly used systems is the traffic alert and collision avoidance system (TCAS), which has been used extensively in civilian aircraft and registered airliners.

Unfortunately, the high cost and extreme complexity associated with the system renders it impractical for use with autonomous flight controller systems and for inexperienced private aircraft owners.

2.2 Reduction of Flight Separation

As discussed earlier, ICAO has introduced new regulations that reduce the nominal altitude separation from 2,000 feet to 1,000 feet. This increases the number of aircraft that can fly in a particular volume of airspace. [3]

These new regulations became known as the Reduced Vertical Separation Minima (RVSM). [6]

RVSM is an aviation term used to describe the reduction of the standard vertical separation from 2,000 feet to 1,000 feet between aircraft flying at levels from FL290 (flight level of 29,000 ft.) to FL410 (41,000 ft.), therefore increasing the number of aircraft that can safely fly in a particular volume of airspace. [6]

Historically, standard vertical separation was established as 1,000 feet for flight levels between sea level and 29,000ft, 2,000 feet from 29,000ft to 41,000ft and 4,000 feet for higher flight levels. This standard was defined because the pressure altimeter used in aircraft has an accuracy that decreases with altitude.

However, newer altimeters and aircraft avionics systems can operate more accurately and are more able to maintain a fixed flight altitude. Therefore it became apparent that for many modern aircraft the 2,000ft separation was overly cautious and ICAO reduced the minimum separation to 1,000ft.

Between 2002 and 2004 RVSM was implemented in much of Europe, North Africa, Southeast Asia and North America, and over the North Atlantic and Pacific Oceans.

However, only aircraft with specially certified altimeters and autopilots may fly in RVSM airspace, other aircraft must fly lower or higher than the airspace, or seek special exemption from the requirements.

One obvious problem is that by reducing the space between aircraft, RVSM may increase the number of mid-air collisions and near-collisions.

The NLR (National Aerospace Laboratory in the Netherlands) estimate that the humans in the loop (pilots and air traffic controllers or ATC) are responsible for 85% of all air-traffic accidents. [7]

This is mostly due to the fact that the performance of Air Traffic Management Systems (ATM) is constrained by the number of aircraft a controller can track simultaneously. Increasing the number of aircraft as per the ICAO recommendations, would ultimately increase the workload of ATC personnel and increases the possibility of catastrophic errors. [7]

ICAO's most important recommendation is to automate the flight controls of an aircraft as much as possible, and to introduce advanced flight computers or auto pilots capable of taking and advising risk reduction decisions.

2.3 Introduction of Free Flight

A new concept of a flexible and changeable flight path emerged during the last decade, and has become known as *Free Flight*.

Intended to replace current air traffic management methods, true free flight eliminates the need for constant ATC command by giving the responsibility to the pilot in control. This gives the pilot the ability to change trajectory in mid-flight. With the aid of computer systems and/or ATC, pilots will be able to make more flight path decisions independently. As in most complex systems, distributed, cooperative decision-making is believed to be more efficient than the centralized control characterized by the current mode of air traffic management. [8]

The many benefits achieved by following a free flight plan, is that a reduction of 4.5% in flight time (and eventually fuel use) can be achieved. This is a reduction of about 500 hours per day in flight time, which translates to nearly one million dollars per day in saved costs.

In addition to this, freeing aircraft from the confined paths of a structured routing system would enhance safety by distributing aircraft flight paths more widely, lessening the dependency on ATC personnel and ground radar and spreading the aircraft more efficiently.

During their research for trajectory conflict solutions, Jardin et al. estimated that for every 3000 aircraft in the air there exist only 40 - 41 collision possibilities that might occur. [9]

However, this also indicates that the possibility of a collision still exists, and the probability of collision changes depending on atmospheric conditions, visibility, pilot experience, efficiency of ATC personnel, and the standards and regulations followed (whether domestic or international flights).

Free flight works by dynamically allocating segments of the airspace to different aircraft at different times and automatically ensuring the required separation between aircraft. However, this requires total reliance on airborne electronic equipment to detect possible collision and, if necessary change path to avoid a collision [1]. Equipping the pilot with an onboard system that can estimate and safely change the flight path would provide an optimum solution to the free flight shortcomings.

The strategic real time optimization of flight-routes requires that conflict-free optimal trajectories are computed on time scales of thirty minutes or more into the future. The trajectories computed would follow a four dimensioned control plan. This is in contrast to the tactical optimization methods where aircraft resolve conflicts as they arise on a scale of 10 to 15 minutes, without considering the longer-range consequences on the eventual trajectory and flight plan changes. [5] & [10].

2.4 Introduction of UAVs into Commercial Use

Piloted aircraft are not the only type of aircraft competing for airspace. The evolution of UAV technology and the capabilities of unmanned aircraft systems (UAS) enable such systems to fly at extremely high altitude and to fly for thousands of kilometres with operational endurance of more than 40 hours for some systems. This enables them to fly at altitudes and speeds that make them comparable with manned civilian aircraft.

Another advantage of UAV systems is the low cost of operating and maintaining these systems compared with conventional aircraft.

Such capabilities have contributed to their wide adoption by the military around the world in various demanding roles, from signal intelligence, to long-range reconnaissance and extended endurance weapons platforms. This success in the military has attracted the interest of civilian markets for similar applications, including patrol, border control, traffic control, cheap and easily maintainable communication link stations, and even next generation commercial airliners.

However, gaining access to the civilian markets requires meeting stringent safety standards and regulations, in order to operate safely in non-restricted airspace.

In a report to the US Department of Defence (DOD), Major Weatherington outlined the need for military and future civilian UAV to follow the flight rules and regulations set up by the Federal Aviation

Authority (FAA) for flying in the National Air Space (NAS) or any civilian/commercial airspace for that matter. This includes the need to have an airborne sense and avoid capability. [11]

The National Airspace System (NAS) is the collection of procedures, regulations, infrastructure, aircraft, and personnel that compose the national air transportation system of the United States. Similar bodies exist in the European Union, Asia, and the Pacific region. However the FAA was selected as an example due to its support to such UAV flight plans and its support to research, NASA, and the aerospace industry for this purpose.

This report outlined that all category III aircraft must comply with article 14 CFR 91.113 these include airliners & HALE UAVs, and that "Sense and Avoid system needs to find & avoid traffic conflicts within ±110º Azimuth measured from longitude axis and ±15º in elevation from the cruise speed level line". [12]

Safety is a primary concern to any civilian aviation operator, and although UAV systems have performed well in the military sector, the design philosophy behind military systems are different to those of civilian market aircraft.

Over the years, UAV have excelled at militarized roles such as battlefield surveillance, long endurance weapons platforms, and long range reconnaissance, all of which intend to reduce the possibility of loss of human life (pilots) to enemy ground fire. Therefore, their design concepts focused on durability, ease of maintenance on the battlefield, modularity, and ease of operation (so that they can be operated efficiently with minimal training), where as operating costs, system efficiency, and user interface were less important. [8]

Unfortunately, such design concepts will not succeed in the civilian aerospace industry, as system efficiency, low cost of operation, and safety records are of the utmost importance.

One other important factor is the purpose of operation, where a military aircraft (unmanned or operated by a pilot) operates primarily as a payload delivery platform with the payload being weapons, bombs, reconnaissance pods, electronic jamming equipment or signal intelligence.

Therefore, cost, weight, function, and performance of UAVs have traditionally been the primary concerns, but not over all system efficiency. Naturally, given its high-risk missions and experimental nature, little attention was paid to system redundancies or other design considerations aimed at increasing reliability, whereas civilian airliners are designed for safety, civilian comfort, reliable service, and efficient flying characteristics.

Under certain conditions the current rules governing unmanned aircraft operation permit UAVs to perform limited tasks in commercial or civilian airspace, but this permission is granted certificate of authorization (COA) for individual singular cases rather than as a general authorization. [13]

The process was originally designed for non-routine military UAV operations in civil airspace. It is therefore lengthy and insufficient when applied to many civil operations, requiring detailed review and approval by FAA authorities for each individual flight to be conducted in the NAS.

The issue of a COA to fly in non-segregated airspace would include the evaluation of the following:

- Detailed description of the intended flight operation including the classification of the airspace to be utilized
- UAV physical characteristics
- Flight performance characteristics
- Method of flying and proposed method to avoid other traffic
- Coordination with ATC procedures
- Communications procedures and systems
- Route and altitude procedures
- Lost link/mission abort procedures
- A statement from the manufacturers and operators of the UAV concerning its airworthiness

The process requires a long lead-time for approval and extensive planning prior to a UAV mission. As an exception process, it also forces a very conservative approach to ensuring safety, often limiting the area of operation and requiring adherence to a pre-determined flight path.

2.5 Deficiencies in Current Systems

The principle behind TCAS is a combination of ATC radar monitoring and an airborne transceiver module known as TCAS.

TCAS is a radio-based transceiver that broadcasts the aircraft identity code, altitude, speed, and heading, when it is in range of other aircraft transceivers. [14]

The system is used as a backup or secondary alert system in the cockpit for pilots, to alert them of any possible intersections or close proximity flight. These alerts are known as ***traffic advisories*** (TA), and in the latest generation of TCAS (versions II & the abandoned III), the system would provide limited collision avoidance instructions (almost always by advising a change of altitude). These instructions are known as ***resolution advisories*** (RA). [14]

The input from the ATC to the pilot concerning changing in altitude, speed and heading, are intended to provide for avoidance measure against collisions.

However, the provision of TCAS equipment in the cockpit of an airliner may not provide enough means to avoid disaster. During the aftermath of a tragic midair collision that claimed the lives of 71 passengers, BFU (the German agency responsible for air accident investigations) found that human factors played a greater role than technical factors in the collision. [9]

The report from their investigation suggested the existence of many deficiencies in the combined ATC-TCAS model, whereby conflicting commands to avert the collision may be issued to the pilot from the

ATC and the TCAS unit. The collision occurred even though after the initial RAs, the TCAS units in both aircraft recognised the worsening situation and issued further urgent instructions to "increase climb" and "increase descent" just before the collision, because the crew of one airliner decided to follow the ATC instructions rather than trust the TCAS system.

Since its inception in the early 1990's, TCAS has been intended purely as a last line of defence. The air traffic control system, complemented by procedural compliance and the maintenance of situational awareness by aircrew, is the first line of defence. It incorporates estimation algorithms designed to detect if the aircraft is getting into a situation that could lead to a collision. It also provides the primary defences for recovery if a collision is imminent. It is only when both the ATC control and recovery defences fail or are breached that TCAS instructions are followed. [15]

The main operation method of TCAS is dependent on peer-to-peer communication via airborne transponder systems. Each aircraft is equipped with a certified transponder module, which when interrogated by TCAS or automatic dependent surveillance – broadcast systems (ADS-B), broadcasts the aircraft's identity code (squawk) and its current altitude on $1030MH_Z$ and $1090MH_Z$ carrier signals respectively. [7], [4], & [14]

The airborne computer calculates the horizontal distance to the intruder from the signal response time, and extracts heading and altitude information from the broadcast messages. If the intruding aircraft is equipped with a Mode A/C or Mode S transponder, TCAS can determine its altitude. The computer calculates the rate of closure from the response times of several signals and calculates the time to the closest point of approach [16]

One of the major limitations of the system is that RAs can only instruct the pilot to climb or descend to avoid a collision (RAs and TAs operate only in the vertical plane), with no actual reference to the required safe altitude.

Although TCAS systems have been successful in preventing catastrophic collision incidents, they are large, complex, and expensive to install on aircraft other than commercial airliners. Helicopters and private light aircraft are not required to have TCAS systems installed on board. A high percentage of these aircraft fly without such systems TCAS systems on their aircraft, and the slow adoption of TCAS has resulted in the older generation of TCAS being installed in aircraft creating incompatibility issues. [17]

In addition, in some instances TCAS operates independently of ATC, as there is no direct communication link between on board TCAS systems and ground radar. [15]

From an operational perspective, TCAS systems rely on the fact that the pilot has the necessary skill to evaluate the danger and respond accordingly, rather than use an automated pilot system to take over and perform the evasive manoeuvre.

Several tests have shown that TCAS communicators are prone to interference and jamming which could result in catastrophic events [17]. Furthermore, numerous reports have shown that conflicting avoidance instructions have been reported amongst pilots and ATC personnel when dealing with aircraft in high-density areas such as airports. [18]

TCAS does not generally issue any RAs if the aircraft is flying at an altitude below 2000ft, but rather the regulations mandate that the pilot should depend on ATC personnel and the aircraft's altimeter at that low altitude. [17]

From the above discussion, it is obvious that one of the major issues is the effect of pilot interaction with the TCAS instructions, where any delay or hesitation by the pilot could result in catastrophic errors. This problem is further compounded by the occasional differences between the ATC personnel and TCAS instructions, which could affect the pilot's reaction time significantly.

Another issue to be addressed is the possibility of a simpler system that can be installed on smaller non-commercial aircraft to provide the same collision counter measures available to larger aircraft.

The introduction of ADS-B will provide more accurate information than TCAS from transponder returns alone, but ADS-B does not provide RAs. These technologies are expected to be complementary to the already existing ATC system.

Some researchers, airline pilots, ground crew engineers and ATC personnel, believe that this would lead to even more confusion and increasing of scepticism of pilots towards the combined systems. [6]

2.6 Limitations of TCAS

Since the introduction of TCAS Systems, pilots and aviation authorities have identified several limitations in the operation of TCAS Systems. [10], [12], & [19]

Some of these shortcomings can be listed as follows:

Dependence on active transponder equipment: TCAS relies upon information received from active airborne transponders, thus it cannot evaluate any aircraft that does not have matching TCAS units

Limited accuracy: Due to the effects of signal degradation and interference, TCAS is incapable of resolving collision situations accurately

False Traffic and Resolution Advisories: Even under optimum operating conditions, TCAS may issue false collision danger reports due to deficiencies in the equipment or received data.

Limited Resolution Advice: Even upon providing resolution advisories, TCAS information is restricted to single command lines such as: Climb, Adjust Vertical Speed, Descend etc**,** which may prove useless in attempting to manoeuvre in a constricted airspace

Another important limitation is the dependency of TCAS on the pilots' response to the issued RA's, whereby TCAS wasn't designed to communicate directly to an autopilot system or directly change to the flight path without human/pilot intervention.

This is unsuitable for UAV operations, due to the lack of a human pilot capable of making avoidance manoeuvres, the concept of re-routing the traffic advisories back to the ground monitoring station (which could be hundreds of kilometres away), where a human ground operator has to assess the danger, estimate a safe solution, and over-ride the automated controls of the UAV to correct the path, is of very limited use.

Figure 1 demonstrates how two civilian jets would try to avoid each other according to TCAS instructions [4].

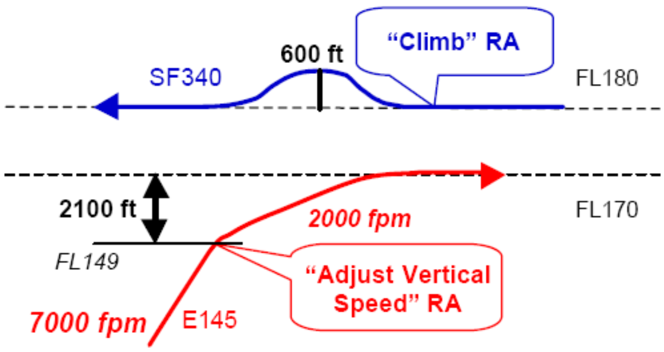

Figure 1 - Evading a collision as per TCAS instructions [4]

One of the primary limitations of TCAS is the limited number of resolution advisories that the system can provide. In the initial version, the system can only provide traffic advisories (i.e. danger of collision) however it cannot provide any further details concerning the relative location of that aircraft nor any resolution advisories (collision avoidance instructions) to that specific threat.

It is also worthwhile to note, that TCAS is a transceiver-based system. i.e. TCAS will only detect and classify other aircraft that carry the same version of TCAS transceivers on board. This greatly limits the system to operate only on civilian airliners. It is completely inadequate for light private aircraft (due to expense and different levels of pilot competency) and UAVs (due to the absence of pilots) in the aircraft. [13]

For these reasons many aviation experts consider TCAS to be outdated technology for modern day collision avoidance systems, and although other more advanced variations have been developed to address some of these limitations (such ADS-B) an advanced active non-cooperative system is needed for UAV applications. [15]

The main focus in recent years has shifted to providing aircraft (whether manned or unmanned) with a universal (or some standardized form) sense and avoid systems.

Sense and avoid systems are based on active and semi-active detection measures such as radar, synthetic aperture radar systems (SAR), ADS-B (semi-active), and/or optical detection type systems. Each aircraft is able to scan a volume of airspace and estimate if there is a collision probability with other aircraft or objects in that airspace, and is able to provide or invoke collision avoidance measures.

To enable UAVs to enter national non-restricted airspace, aerospace companies would have to demonstrate advanced measures to enable ground crew and UAV autopilots to sense and avoid other aircraft.

Raytheon's ***Universal Control System***™ (UCS) addresses the remote human-machine interaction by investigating similar controls from the video-gaming industry. The billions of dollars spent in that sector have resulted in the development of simple, intuitive, and instinctive interfaces. [19]

The validating of sense and avoid systems would allow UAVa to operate more quickly and cheaply by flying direct routes to their destinations.

Also, Raytheon suggests that such sense and avoid technologies can be applied to ATC ground crew and commercial airliners to provide better situational awareness and reduce human mistakes that contribute to approximately 67% of all aerial accidents. [19]

In this project, we shall concentrate on an alternative system that can be used on UAV and can be easily converted to light aircraft, to compensate for the inexperience of amateur pilots and provide them with resolution advisories even if they are outside the range of ATC communications.

CHAPTER 3

Design Specification and Requirements

3.1 System Requirements and Operational Specifications

Current rules governing UAV integration into civilian airspace, state that the design criteria for a UAV must take into account its intrinsic safety, mode of operation, and the environment in which it operates.

Specifically, the rules apply to the ***Sense and Avoid*** (S&A) systems, to be installed on UAVs to provide them with the ability to detect, evaluate, and avoid if necessary other aircraft in the shared air space.

According to European Organisation for the Safety of Air Navigation (EUROCONTROL), a UAV S&A system should enable a UAV pilot-in-command to perform those separation provision and collision avoidance functions normally undertaken by a pilot in a manned aircraft, and it should perform a collision avoidance function autonomously. The S&A system should achieve an equivalent level of safety to that of a manned aircraft. [20]

In essence, any S&A system should provide the ability to detect conflicting traffic in time to perform an avoidance manoeuvre. The system would then notify the ground crew monitoring the operation of the UAV of the conflict and propose a course of action to pass well clear.

In the subsequent event of inaction or absence of override by the UAV pilot-in-command, the S&A system would manoeuvre the UAV autonomously (change speed, altitude, and heading) to avoid the conflicting traffic. [21]

Therefore it is paramount that an accurate and a reliable collision avoidance system would require the UAV to navigate, sense, and avoid other aircraft within the planned operational course in order to provide the same level of safety as a manned-aircraft.

In a recent study, NASA in association with the US Air force and the FAA recognized that in order to allow unmanned aircraft to fly in unrestricted airspace, the specifications for the collision avoidance system should follow three basic principles [22]:

- UAV operations should not increase the risk to other aircraft
- Air Traffic Management and avoidance procedures should mirror those applicable to manned aircraft
- The provision of air traffic services to UAVs should be transparent to Air Traffic Controllers.

Furthermore, EUROCONTROL in cooperation with the FAA and other leading aviation authorities has identified certain operation safety parameters for the S&A system, whereby the UAV or the pilot-in-command is responsible for maintaining separation at all times, i.e. the UAV must allow for right of way for other civilian aircraft sharing the same airspace. The UAV must also maintain an approximate distance of 1000ft horizontally or 500ft (approximately 150m) vertically between the UAV and other airspace users. [21]

In the case of the "Sense and Avoid" Project, the system specification have been established as follows: the system specifications of the sense and avoid project are to design and develop an airborne collision avoidance system that relies on a SAR system to detect aircraft approaching the UAV and to perform a manoeuvre to avoid colliding with any of these aircraft and establish a minimum separation of 1000ft (approximated to 400m) in all directions. This minimum separation value allows for errors due to detection inaccuracies and wind shear effects.

3.2 Regulations Concerning Collision Avoidance Systems

As per the S&A functionality definition, the UAVs must have a "Detect, Sense and Avoid" system that allows them to detect and safely change paths in order to avoid aircraft or other obstructions.

For UAVs, the FAA states that UAVs operating in a civilian airspace must provide an equivalent level of safety, comparable to the sense and avoid requirements for manned aircraft and in compliance to CFR 91.113 (Code of Federal Regulations number 91.113). [23]

These specific flight regulations govern the "Right of Way Rules" for manned aircraft in flight, specifically the rules that dictate what manoeuvres pilots should follow to avoid a collision and specifically notifying ATC personnel concerning the change in altitude. In particular, the regulation specifies that an aircraft that is overtaking or changing heading in order to avoid a collision, must turn to the starboard (right) side of its flight direction.

Since the intention of this thesis is to provide a simulation for a S&A system that can be applied to both manned and unmanned aircraft sharing the same non-segregated airspace, it is necessary to follow these flight rules to allow for homogeneous collision avoidance manoeuvres for all aircraft.

This proposal was confirmed by the adoption of such rules by NASA throughout their ERAST project [24], and by Panicker et all [25] in their investigation of collision avoidance strategies.

These rules were to be adopted by the US Air Force in studies conducted by Major Weatherington (US Air Force) [11] and the Defence Sciences Study Board [26] notes on integrating unmanned aircraft into civilian airspace.

To satisfy the requirements, all UAVs must therefore be able to reliably avoid collisions with all aircraft (cooperative and non-cooperative) at all times. [13]

The FAA requirements discussed above are described in detail in a multitude of documents and sources for pilot operated aircraft. However, defining "equivalent level of safety" for see-and-avoid is still unclear. In fact, no specific rules exist for the operation of UAVs.

However, current regulations and proposals agree that for an equivalency standard, unmanned aircraft must have a detection range of an azimuth angle of ±110 degrees and elevation of ±30 degrees. In addition to that, the F38 Committee (the name of the committee responsible for regulation of future flight rules for UAVs) has issued further collision avoidance regulations that state any UAV flying in a civilian airspace, must be able to respond and avoid collision by a separation distance of at least 500ft (approximately 150m). [27]

Nevertheless, this could be an insufficient standard. FAA data indicates that most midair collisions occur in clear daylight conditions when a faster aircraft is overtaking a slower aircraft.

Limitations in the rear aircraft detection capability of the slower aircraft are part of the cause. In a UAV, aircraft detection capabilities need not be restricted to forward looking capabilities but can have a 360 degree viewing range depending on sensor type and placement. [27]

3.3 Aim of this Research

The current aim of this experiment is to develop and simulate a collision avoidance system for a medium sized unmanned helicopter called the Snark. This S&A system will interface to the aircraft flight computer and to the radar module on the flight deck.

The S&A system will operate as an independent collision evaluation unit, reducing the processing load on the main flight computer, and the system will send warning messages to the flight computer (these would eventually be transmitted back to the remote pilot of the aircraft) to indicate a proximity breach and with the possible solution to avoid the collision. Furthermore, the S&A system would instruct the flight computer/autopilot to change course if no action was undertaken.

The main goal of the simulated tests is to qualify the component as worthy for further controlled/limited flight tests on the aircraft once the airframe is airworthy.

The test would involve simulating the radar system, and the detection of other aircraft within range of the radar. The test would also verify how the avoidance mechanism would calculate whether the paths of the two aircraft would intersect or if the two aircraft would be at any point closer to each other than the minimum separation distance.

In the case of a possible collision or near miss paths, the software would calculate a different path for the UAV to undertake.

All of these calculations will use the the motion equations to provide a more realistic system and UAV response time to the situation.

The aim of such a simulation is to prove that this research can be used as a building block for more extensive hardware systems that can be installed on experimental aircraft or full scale UAVs for future airworthiness testing.

3.4 System Architecture

In the heart of the S&A system, lies the logic engine responsible for generating the critical decisions for changing the UAV’s course or speed, or both in order to avoid a collision with another aircraft. The logic engine should satisfy two conditions:

- Detecting when a collision becomes a possibility due to the predicted separation between the Snark and the approaching aircraft being less than the minimum allowed safe separation
- Finding the minimum change in velocity and/or flight path to enable the Snark to avoid a potential collision, while deviating as little as possible from the original flight plan.

The architecture of the S&A system comprises of a logic unit which will compute the possibility of a collision with another aircraft and will determine the avoidance action. This unit will communicate with the flight controller unit (or the Autopilot) and the airborne radar unit.

According to the actual physical layout of the aircraft avionics systems, each of the radar, S&A system, and flight controller units will be based in separate compartments throughout the aircraft.

The main requirement of the project is to encapsulate the unmanned helicopter system in a safe-zone bubble that is 400m in radius, where the intention is provide this buffer zone between the aircraft and any obstacle in its path.

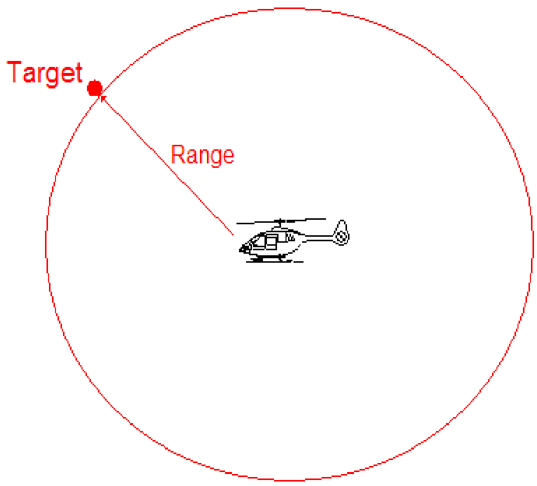

Figure 2 - Target detection/Safety separation zone

CHAPTER 4

AIRCRAFT SIMULATION MODELS

4.1 Point Mass Aircraft Model

In order to achieve a realistic response during the simulation and testing stages of the collision avoidance algorithms, a dynamic model for the aircraft must be selected to represent the aircraft's response and instantaneous velocity and directional vectors.

The ***Point Mass*** aircraft model captures most of the dynamic effects encountered in civil aviation aircraft. This type of modelling is very popular in the estimation of aircraft performance, and can be directly implemented in the collision detection and avoidance algorithms [23].

The main characteristic of the ***Point Mass model*** is that the aircraft's thrust is directed along the main velocity vector, and that the aircraft always performs coordinated manoeuvres. Another advantage of the ***Point Mass model*** is that at detection ranges under 200 nautical miles (where most conflict scenarios and their resolutions occur) no need exists to compensate for the earth's curvature, nor for any lags or delays induced by its rotation.

This would be extremely beneficial for the simulation purposes in this project, as there will be no need to introduce complex model formulae, and the behaviour of the aircraft (or UAV in this case) can be represented by simple Newtonian dynamics as follows [28].

The aim of this research is to estimate the best response to avoid a collision, where the collision avoidance unit will operate as a subsystem of the flight controller module (which will automatically adjust thrust, turning coordinates, acceleration values, and engine/rotor output power to climb or to descend).

$$V_X = V_0 \times Sin\alpha \times Cos\beta$$

$$V_Y = V_0 \times Cos\alpha \times Cos\beta$$

$$V_Z = V_0 \times Sin\beta$$

Where

Vo is the velocity

V_X is the Horizontal Velocity

V_Y is the Forward Velocity

V_X is the Vertical Velocity

Where α is the horizontal travel angle and β is the vertical elevation angles

Please refer to Figure 4 for further details

Equation 1 - Point-Mass velocity vector equations

Thus the resulting algorithm can be easily adapted to different airframes (aircraft types) with little direct dependency on aerodynamic performance.

As part of the ***Point Mass Model***, the use of trajectory parameterisation techniques is the key to the operation of the collision avoidance algorithm. The path trajectories of all of the individual aircraft detected by the radar system can be estimated using a limited number of parameters.

In this project, we use a simple piecewise linear trajectory to approximate the movements of both the UAV and the other aircraft, during the processing stages of the aircraft path trajectories, and during the collision warning estimation and evasion tactics.

However, to simulate a more natural change in the Snark's path or velocity, we would need to simulate the trajectories and aerodynamic performance of the Snark. This would achieve smooth and accurate trajectories and path changes.

4.2 The Snark UAV

This research project is intended to provide a simulation model and an applicable software base to form a collision avoidance system for the Snark UAV, which will perform as a test aircraft for the system.

The Snark Unmanned Aerial Vehicle was an experimental unmanned helicopter platform proposed by TGR Helicorp. It was intended to be a light search and rescue, or autonomous rotary aircraft, with medium-high altitude capabilities and long flight endurance.

Certain aspects of this platform's performance are classified, however, we have been allowed to elaborate that it is an unmanned helicopter/rotor craft capable of high altitude/long endurance flight, with an estimated maximum dash speed of 145 knots and a cruising speed of 100 knots.

One of the intended operational requirements of this aircraft is the ability to fly in integrated airspace. Therefore a collision avoidance system must be added to the system to enable safe operation.

The collision avoidance system devised in this project can be applied to all platforms of aircraft, manned or unmanned. However we shall assume that the initial prototype of the system (and the simulation model) will be based on a helicopter platform as per the performance specifications of the Snark UAV, as it is the initial test aircraft.

4.3 Analysis of Helicopter Flight

An important consideration for this report is how to model the movement of a helicopter in flight, i.e. what are the major flight rules, velocity equations, and manoeuvre techniques that apply to helicopters during flight.

From an aerodynamic point of view, helicopters modes of flight differ a great deal from those of a standard fixed wing aircraft. Unlike fixed wing aircraft, helicopters do not depend on varying lift forces acting on the wings, to climb or descend, nor on changing the flaps, rudder location, or ailerons to change direction. Instead, helicopter flight largely depends on the amount of lift force generated through the rotation of the blades around a central axis.

Helicopters rely on changing the pitch angle of the rotating blades to achieve lift forces vertical to the plane of rotation of the rotors (i.e. normal to the rotating disc drawn by the spinning rotors). In addition, the forward direction and velocity are directly related to the offset angle of the vertical lift force to the vertical axis of the main rotor hub. [23]

A typical helicopter has three separate flight control inputs. These are the cyclic, the collective, and the counter-torque pedals. The cyclic control changes the pitch of individual rotor blades depending upon its position as it rotates about the rotor head. The result is to tilt the rotor disk in a particular direction, resulting in the helicopter moving in that direction.

The collective control on the other hand changes the pitch of the rotor blades concurrently regardless of their position. Therefore, if a collective input is made, all the blades change equally, and the result is the helicopter increasing or decreasing in altitude

However despite having such a complex aerodynamic model, Prouty argued that a helicopter displays the same flight control responses and behaviour as those of a fixed wing aircraft when it is travelling at speeds in excess of 15kts (or 28km/hour) in forward flight. [29], [30]

The mathematical model used throughout this research and any collision avoidance measures simulates the Snark in forward flight thus behaving similarly to a fixed wing aircraft comparable in size.

A clear requirement for pilots to maintain safe flight is the ability to predict the future trajectory of their aircraft far enough ahead that they can stop, turn or climb to avoid a hazard. One such collision avoidance strategy is to control directly, the approach velocity to the target. [31]

4.4 Radar System Simulation

To avoid other aircraft it is imperative to detect the other aircraft flying within a predefined detection range, the challenge lies mainly in meeting the performance requirements while keeping cost, size and weight to a small portion of the total payload so that UAVs can carry out their intended mission.

During the experimental phase of the ERAST project led by NASA and Amphitech, Bernier et al identified that the radar should be able to provide the UAV non-cooperative aircraft detection capability with sufficient time to identify, assess and take action in accordance with the situation encountered. [32]

Throughout the ERAST project, NASA identified that the optimum range of detection is 8 nautical miles (approx 14.8km). Unfortunately, light airborne-radar systems that can detect flying aircraft at distances of hundreds of kilometres do not exist yet. However, new radar systems, specifically designed for light unmanned aircraft with limited payload capacity have been designed with dual use as ground surveillance radars and airborne warning systems. The most suitable of these systems include:

- MiSAR Radar developed by EADS Electronics and currently used on the German Army's light unmanned reconnaissance aircraft (Aladin)

- Lynx Radar System from Sandia National Laboratories
- OASys Radar System developed by Amphitech and used for the evaluation of the NASA ERAST project
- Unicorn Airborne Warning System from Flight Safety Technologies

The Lynx and OASys Systems being the most powerful can detect targets at distances of 20-30kms, while the MiSAR and Unicorn units, being lighter and designed for lighter unmanned aerial systems can detect targets at ranges of approximately 10-12kms.

The first three radar systems operate in the Ku-Ka bands with frequencies varying from 15GHz to 35GHz. The compactness of this system depends largely on their design being based on Synthetic Aperture Radar scanning schemes, which in addition to their extremely high frequency signal help reduce the size and power requirements for the transceiver circuits and component sizes, reducing overall unit weight [26], [27], and [29].

Over the past few years, multiple experiments including close coordination of air traffic controllers, FAA, CAA, and EUROCONTROL have shown that the time gap between two approaching civilian aircraft should be no less than 20 seconds. This gives just enough time for the pilot to identify where the threat is, consider the safest action to take, and to complete that manoeuvre [32]. For two modern airliners with cruising speeds of approximately 600kts, this 20 second gap can be calculated to approximately 6.2 km.

And according to international flight rules, closing speeds for altitudes below 18,000ft (5.5km) should be approximately 500 knots this in turn would give approximately 21 seconds to avert a collision. [33]

Furthermore, the radar system threshold revisit or scan rate is set to 1 Hz. The rationale behind fast revisit rates is that, for the purpose of detecting a possible collision with another aircraft path, the target must have been tracked over two to three scans. [32] & [12]

From the previous arguments, we shall assume for our unmanned helicopter design that the maximum detection range in an unobstructed environment should be 10km.

Based on the joint project between NASA, Amphitech, and Scaled Composites, Bernier et al set a minimum criteria for the radar system required for airborne collision detection and avoidance system. This includes: [32]

Detection Range	10km
Scan Rate	1 sec
Theoretical Scan Zone	360º
Field-of-Regard in Azimuth	±110º
Field-of-Regard in Elevation	±60º
Probability of Detection	> 90%
False Alarm Rate	<10%

Angular Accuracy	1.5º
Range estimation error	< 76m
Minimum Distance for Detection	5.5km
Minimum Distance for Tracking	5km

Table 1 - Radar system requirements

Unfortunately, the characteristics of the radar system can vary depending on the version of the produced system, the aircraft's power supply, payload capacity of the aircraft, and system cost.

Radar systems are too expensive to be owned by light aircraft owners, and their complexity present a challenge to relatively inexpensive and routine servicing required for such aircraft, as they can only be maintained by specially trained personnel. These factors prohibit them from being used as a standard subsystem on light or private aircraft. Even commercial airliners do not carry active primary radar systems on board but rather depend on TCAS, ADS-B, and air traffic controllers for guidance and navigation.

However, recent advances in passive and micro strip patch-array promise to reduce the cost and complexity of efficient radar systems that can be installed on light or large aircraft with minimal power and processing requirements [34]. One such example is the Unicorn detection system, designed by Flight Safety Technologies Ltd. [34]

This system is comprised of faceted micro-strip patch antenna transceivers. These can be grouped together into sub-arrays to provide a 360º of horizontal (Azimuth) detection and 180º of vertical (Elevation) detection per sub-array.

Figure 3 - Unicorn micro-strip patch antenna

For this experiment we assume that the detection system is of a compound-sensor type (multiple radars or active arrays positioned around the aircraft) and would provide:

- 360º coverage in both azimuth and elevation angle (i.e. a spherical coverage around the Snark)

- A detection range of 10km
- A scan rate total detection field of 1 second
- Possibility of detection greater than 90%
- Less than 30m of error value (this error values is significantly less in both the OASys and MiSAR radar systems)
- Capability of detecting and tracking multiple aircraft up to 5 aircraft.

4.5 Target Range and Tracking

Throughout the mathematical derivations, we shall assume that the radar system operates as follows:

The radar identifies a target R distance away (with vertical elevation angle β and horizontal angle α) and the radar will also calculate the relative velocity of the target.

We will also assume that the output from the DSP/radar stage of the system is: ID, (R_X, R_Y, R_Z), (V_X, V_Y, V_Z)

That the axis system used throughout this research assumes that the Snark is traveling horizontally along the y-axis. In this axis system the horizontal plane is formed by the x and y axes, and the z axis is perpendicular to that plane.

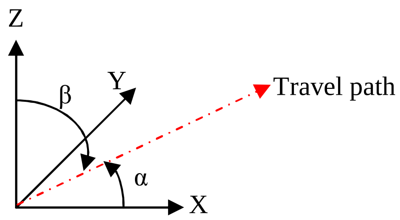

Figure 4 - Representation of coordinate system relative to travel path

In the case of the MiSAR radar system, where it provides just a distance and the angles suggested above, we could compute the distance and vector components using:

$$\begin{aligned} R_X &= R \times Sin\alpha \times Cos\beta \\ R_Y &= R \times Cos\alpha \times Cos\beta \\ R_Z &= R \times Sin\beta \end{aligned}$$

Equation 2 - Calculation of range coordinates in 3D environment

And determine the velocity of the target by calculating the rate of change in the range vector values (R_X, R_Y, R_Z).

We estimate if there is going to be any collision or near collision as follows.

Firstly , a collision is possible if R_1 is less than R_0, where R_0 is the initial distance at which that target was detected for the first time, and R_1 is the subsequent range reading of that target. Hence, the target is getting closer to the flight path of the Snark.

Next, calculate if the target is going to nearly collide with the Snark by using the following set of equations to predict the future target distance:

$$CD_X = R_{0(X)} - V_X \times t$$
$$CD_Y = R_{0(Y)} - V_Y \times t$$
$$CD_Z = R_{0(Z)} - V_Z \times t$$
$$CD = \sqrt{CD_X^2 + CD_Y^2 + CD_Z^2}$$

CD represents the Collision Distance, whereby if that distance CD < 400m at some future time t then there is a collision possibility with that aircraft in t seconds. This will be the prime condition for the Snark to identify a safe velocity or travel path to avoid the potential collision.

The detection algorithms and mathematical formulae assume a flat-earth model. Also, the 400m listed in the equations, represents the minimum separation distance that should be between the Snark and any other aircraft at all times. According to the new European regulation, the separation distance between aircrafts should be 1000ft, or 334m [4]. We designed for a longer safe separation of 400m to allow for radar inaccuracies as discussed earlier.

The flat earth model was chosen because of the short range of detection (10km). Therefore there would be negligible effect of the effects of the earth's curvature on the accuracy of the measurements, which are required to be compensated for if the radar detection range is larger than 30miles. [35]

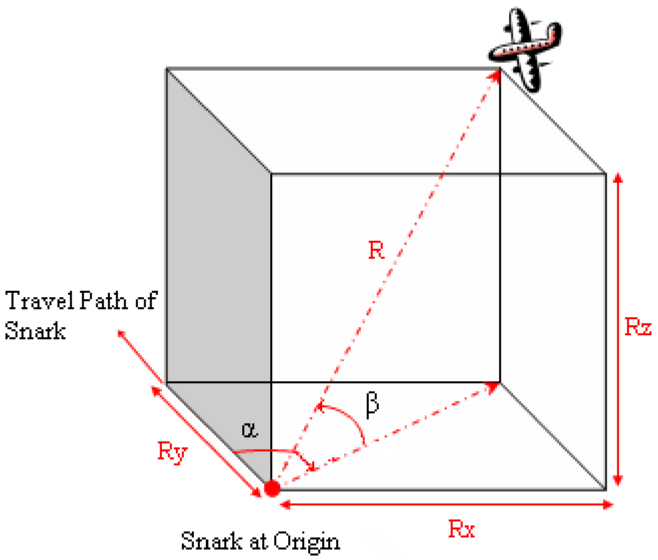

Figure 5 - Distance and velocity vector relative to Snark

4.6 Aerodynamic Characteristics and Manoeuvring Capabilities

A vector analysis approach is used in the collision avoidance algorithm to determine which action the Snark should undertake (i.e. change in velocity or changes in altitude and direction) to avoid a collision.

There are many factors that can affect the ability of the Snark to perform certain manoeuvres. These may include: altitude (affects the turning rate, radii and climbing rates), the load of the aircraft, the nature of the terrain, wind shear, humidity, and fuel levels.

For software simulation purposes and due to the lack of exact technical information regarding the performance of the Snark, we have conducted the simulation with technical specifications of the Robinson R-44 and Schweizer 300c helicopters, which are quite similar to the Snark in:

- Overall dimensions (R44)
- Gross Weight (R44)
- Rotor construction, number of blades, and power to weight ratio (Hughes/Schwarz 300c)
- Engine output power and performance (both models)
- Loading factors and effects of carried stores on the performance (R44)
- Manoeuvring and hovering capabilities, G-ratings (Hughes/Schwarz 300c)

Although the R-44 is a civilian helicopter without the intended loading flexibility and advanced design metrics of the Snark, the similarity in size, weight and aerobatic performance is sufficient for the purposes of this research.

The following represent the performance characteristics attributed to the Snark during this simulation: [36] & [37]

- Cruising speed is 100kts
- Maximum straight level speed 145kts
- Hovering ceiling with maximum operational load is 10,600ft with 17kt cross wind
- Rate of climb with maximum load is 5m/sec for altitudes below 10000ft and 3m/sec for altitudes above 10000ft

In addition, the Snark has been designed in accordance to the Codes of Federal Regulations for Rotor Aircraft, where it can withstand approximately 1.5g forward flight acceleration/deceleration forces and approximately +3.5g to –1.0g rates of turn acceleration forces.

4.7 Operational Modes of the Flight Controller unit

The flight controller selected for the UAV is the UAVNavigation AP04/GCS02 rotorcraft flight control system. The AP04 autopilot is installed in the helicopter and interfaces to the aircraft sensors (accelerometers, gyroscopes, magnetometers, and static and dynamic pressures), and a GPS locator. [38]

The GCS02 ground control station relays all UAV communications to UAV Navigation GCS software both in the downlink and uplink directions. Additionally, the GCS02 can provide a manual control for a UAV directly from a standard remote control style joystick connected to the GCS02. [38]

This flight controller was selected due to its small physical size, minimum power requirements, and multiple flight modes. This specific flight controller module is designed to operate aircraft autonomously and control the heading, airspeed, and altitude according to either a preloaded flight plan or to real-time input from the pilot on the ground station. Furthermore, the software can be modified to allow for in-flight course (heading and altitude) and airspeed changes. In this research it is envisaged that the main flight computer of the aircraft will be controlled by the instructions from the collision avoidance program.

The AP04 autopilot has several operational modes, which are selected depending on the type of mission and the frequency of command inputs received from the ground station. These include: MANUAL, DIRECTED, AUTO, FLYTO and HOVER modes.

In the MANUAL mode, the aircraft is operated under full control of the pilot on the ground station (or in our case, under the control of the collision avoidance system).

In the DIRECTED mode, the aircraft maintains a specific altitude, airspeed, and heading as specified by a manual input from the ground station (or the collision avoidance system) and the aircraft would continue to fly under the specified airspeed, altitude and heading until they are changed again by the crew of the ground station.

Under the AUTO mode, the autopilot follows a predefined/preset route, in which the aircraft then will follow the flight plan that is comprised of several waypoints. Each waypoint specifies a set of flight parameters for velocity, aircraft heading and altitude. The Snark will fly towards waypoint(n) under the specified flight parameters until it reaches that waypoint, and then it will change its flight parameters if needed, to those of waypoint(n+1) and so on.

The operator can command the UAV to go to a specific location using the FLY TO mode. Once the UAV reaches the target location, the autopilot switches to HOLD mode and the UAV will execute a HOLD pattern around the target location

Of particular interest to this experiment are the MANUAL, DIRECTED, and FLYTO modes. A combination of the MANUAL and DIRECTED modes is used for a more controlled flight plan, where the program would continuously monitor and update the climb/descent rates, the airspeed, and the heading/turn rate of the aircraft.

The FLYTO mode is used to direct the aircraft to autonomously fly to a specific way point or location coordinates. One important feature of the FLYTO mode, is that the flight controller operating under this mode, may instruct the helicopter to perform high G short turns, especially when changing heading. This occurs because the FLYTO mode is usually used for flying on specific paths, and to fly over specific waypoints, therefore the flight controller would try to fly according to the planned flight path, and will attempt to deviate the least from it, resulting in fuel economy, and better operational performance.

CHAPTER 5

AVOIDING A COLLISION

5.1 Calculation of Minimum Separation

The collision avoidance process consists of two major parts. The first part is the estimation of the collision possibility with another aircraft, i.e. estimating if the separation distance between the two aircraft is under the 400m safety limit. The second part is deducing which collision avoidance manoeuvre will ensure at least 400m-separation between the two aircraft.

A vector-based approach provides the most accurate means of estimating the path of an aircraft and the collision/proximity breach possibility because of that path. Vector based algorithms for determining collision avoidance resolutions are extremely robust, and their performance does not contain any anomalies as with artificial intelligence based systems. In addition, no collision scenario presents any difficulty to the operation of the algorithm in contrast to rule based models. [1]

According to Padfield *et al* [31] one important requirement for pilots to maintain safe flight is the ability to predict the future trajectory of their aircraft far enough ahead so that they can stop, turn or climb to avoid a hazard. This algorithm satisfies this condition as it can almost immediately predict the flight path of an aircraft and therefore be able to invoke a resolution manoeuvre.

A two-dimensional scenario (2D vectors) that demonstrates the functionality of this algorithm is illustrated in Figure 6.

At initial detection, at time zero, an aircraft is detected at a range value of R_0 and an incident angle α (point P_0) as indicated by the figure. It is assumed that the radar system also outputs the velocity (VAS) of the aircraft relative to the Snark.

$$\overline{VAS} = \overline{V}_A - \overline{V_S}$$

Where V_A is the velocity of the aircraft, and V_S is the velocity of the Snark.

The algorithm estimates:

- The closest point of approach for the target (occurs at point Q)
- The value of the minimum distance (d) to the Snark and the time (t_Q) it takes for the aircraft to reach that point Q.

From a mathematical point of view, the behaviour of the incoming target aircraft can be modelled according to its location relative to the Snark.

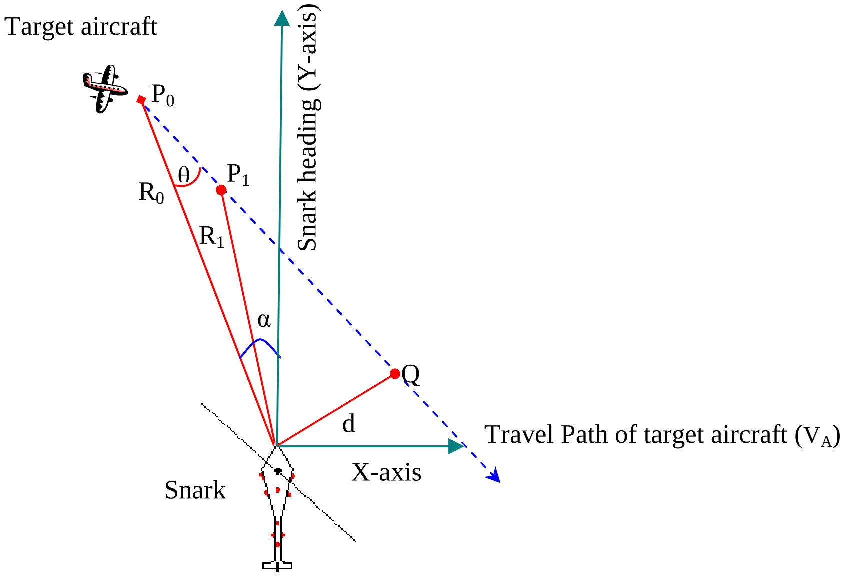

Figure 6 - Closest path estimation

To calculate the value of the minimum distance (d) and the time (t_Q) until the target is at the nearest point to the Snark, we perform the following calculation

$$\overline{A}(t) = \overline{A}_0 + \overline{V}_A \times t$$
$$\overline{S}(t) = \overline{S}_0 + \overline{V}_S \times t$$

Where $\overline{\text{A}}$(t) represents the location of the target aircraft at time t

and $\overline{\text{S}}$(t) represents the location of the Snark at time t

The separation (SP) between the two aircraft is the difference between $\overline{\text{A}}$(t) and $\overline{\text{S}}$(t)

$$\overline{\text{SP}} = \overline{A}(t) - \overline{S}(t)$$
$$\overline{\text{SP}} = (\overline{A}_0 + \overline{V}_A \times t) - (\overline{S}_0 + \overline{V}_S \times t)$$
$$\therefore \overline{\text{SP}} = (\overline{A}_0 - \overline{S}_0) + (\overline{V}_A - \overline{V}_S) \times t$$

5.2 Calculation of Time to Reach Minimum Separation (t_Q)

The magnitude of the distance between the two aircraft can be calculated as follows:

$$Separation\,(SP) = \sqrt{\begin{array}{l}[(A_X - S_X) + (V_{AX} - V_{SX}) \times t]^2 \\ + [(A_Y - S_Y) + (V_{AY} - V_{SY}) \times t]^2 \\ + [(A_Z - S_Z) + (V_{AZ} - V_{SZ}) \times t]^2\end{array}}$$

Squaring both sides of the equation gives:

$$\mathrm{SP}^2 = \left\{ \begin{array}{l} \left[(A_X - S_X) + (V_{AX} - V_{SX}) \times t\right]^2 \\ + \left[(A_Y - S_Y) + (V_{AY} - V_{SY}) \times t\right]^2 \\ + \left[(A_Z - S_Z) + (V_{AZ} - V_{SZ}) \times t\right]^2 \end{array} \right\}$$

Differentiate both sides by time, to gives:

$$\frac{d\, SP^2}{dt} = \frac{d}{dt} \left\{ \begin{array}{l} \left[(A_X - S_X) + (V_{AX} - V_{SX}) \times t\right]^2 \\ + \left[(A_Y - S_Y) + (V_{AY} - V_{SY}) \times t\right]^2 \\ + \left[(A_Z - S_Z) + (V_{AZ} - V_{SZ}) \times t\right]^2 \end{array} \right\}$$

Since the separation value at time t_Q is at its minimum value, this derivative at time t_Q is zero.

$$0 = \left\{ \begin{array}{l} \frac{d}{dt}\left[(A_X - S_X) + (V_{AX} - V_{SX}) \times t_Q\right]^2 \\ + \frac{d}{dt}\left[(A_Y - S_Y) + (V_{AY} - V_{SY}) \times t_Q\right]^2 \\ + \frac{d}{dt}\left[(A_Z - S_Z) + (V_{AZ} - V_{SZ}) \times t_Q\right]^2 \end{array} \right\}$$

$$\Rightarrow \left\{ \begin{array}{l} 2\left[(A_X - S_X) + (V_{AX} - V_{SX}) \times t_Q\right](V_{AX} - V_{SX}) \\ + 2\left[(A_Y - S_Y) + (V_{AY} - V_{SY}) \times t_Q\right](V_{AY} - V_{SY}) \\ + 2\left[(A_Z - S_Z) + (V_{AZ} - V_{SZ}) \times t_Q\right](V_{AZ} - V_{SZ}) \end{array} \right\} = 0$$

Expanding and solving for t_Q gives:

$$t_Q = -\frac{(A_X - S_X)(V_{AX} - V_{SX}) + (A_Y - S_Y)(V_{AY} - V_{SY}) + (A_Z - S_Z)(V_{AZ} - V_{SZ})}{(V_{AX} - V_{SX})^2 + (V_{AY} - V_{SY})^2 + (V_{AZ} - V_{SZ})^2}$$

Since all of the calculation is performed with the location of the Snark located at the point of origin, and the location of the target aircraft is compensated by using the values of Range and Azimuth angle (α) at time t:

$$\boxed{t_Q = -\frac{R_{0X}\, V_{ASX} + R_{0Y}\, V_{ASY} + R_{0Z}\, V_{ASZ}}{V_{ASX}^{\;2} + V_{ASY}^{\;2} + V_{ASZ}^{\;2}}}$$

Equation 3 - Calculation of time to reach minimum separation (t_Q)

5.3 Calculation of Minimum Separation

From Equation 3, we can determine the time required by the aircraft to reach point Q (point of minimum separation) from its current location. The minimum separation (d) will occur at the instant when the vector (d) is perpendicular to the travel path of the aircraft:

From this, an formula can be derived to calculate whether the incoming aircraft will pass through the 400m safe-zone (the virtual bubble surrounding the Snark), from Equation 3. Where the minimum separation distance (d) is:

$$
\begin{aligned}
d_X &= R_{0x} + V_{ASX} \times t_Q \\
d_Y &= R_{0Y} + V_{ASY} \times t_Q \\
d_Z &= R_{0Z} + V_{ASZ} \times t_Q \\
&and \\
d &= \sqrt{d_X^2 + d_Y^2 + d_Z^2}
\end{aligned}
$$

Equation 4 - Calculation of separation distance (d)

If the value of the smallest separation distance (d) is less than 400m, the avoidance algorithm will initiate a request to manoeuvre to adjust the flight path of the Snark, and increase the separation. The time to collision (t_Q) is the time available for the Snark to complete its avoidance manoeuvre

5.4 Possible Avoidance Manoeuvres

If the minimum separation distance is less than the required 400m, the S&A system is required to evaluate the best action to undertake in order to increase this distance. Several methods have been evaluated in the past to provide autonomous collision avoidance mechanisms for robotic applications.

A prime condition of this research was to develop a system that would follow international flight codes and aviation path changing standards. According to Wall [39], EUROCONTROL has specified 31 rules to enable the integration of UAVs in European civil airspace/air traffic management systems.

These following rules were discussed in the EUROCONTORL document [39], and deal directly with collision avoidance regardless of the type of UAV. These general rules are:

- Ensure that introducing UAVs into civil airspace would not increase the risk of other airspace operations.
- Air traffic Management will be identical for manned and unmanned aircraft.
- Air traffic controllers will handle UAVs in the same manner as manned aircraft.
- In the case of data-link failure, UAVs must be capable of autonomous flight and avoidance.
- The S&A system needs to provide air traffic controllers with a standard and predictable avoidance response.
- Unmanned S&A system should provide the same level of safety as manned aircraft

Therefore the avoidance manoeuvre has to provide a safe and efficient means of resolving a collision incident.

In this research, we have decided to adopt the following collision avoidance actions whenever possible depending on the nature of the collision scenario:

- Change the speed of the unmanned aircraft while maintaining its heading: This often provides a safe, accurate, and the most efficient action to maintain the 400m separation distance.
- Change the direction of the flight path: This action is used to avoid severe scenarios such as a head on collision danger or when allowing right of way to a faster aircraft. Unfortunately it is the most taxing in terms of fuel consumption, and manoeuvre and path recovery times.
- Changing the flight altitude: this action, may be suitable to avoiding collisions when there exists an incursion in the altitude or vertical separation, but it is less likely than the others due to the slow climb and decent rate of the Snark.

These three actions (or a combination of them) can be easily applied to avoid a single or two aircraft that are on collision paths, as will be discussed and proven in the next few sections.

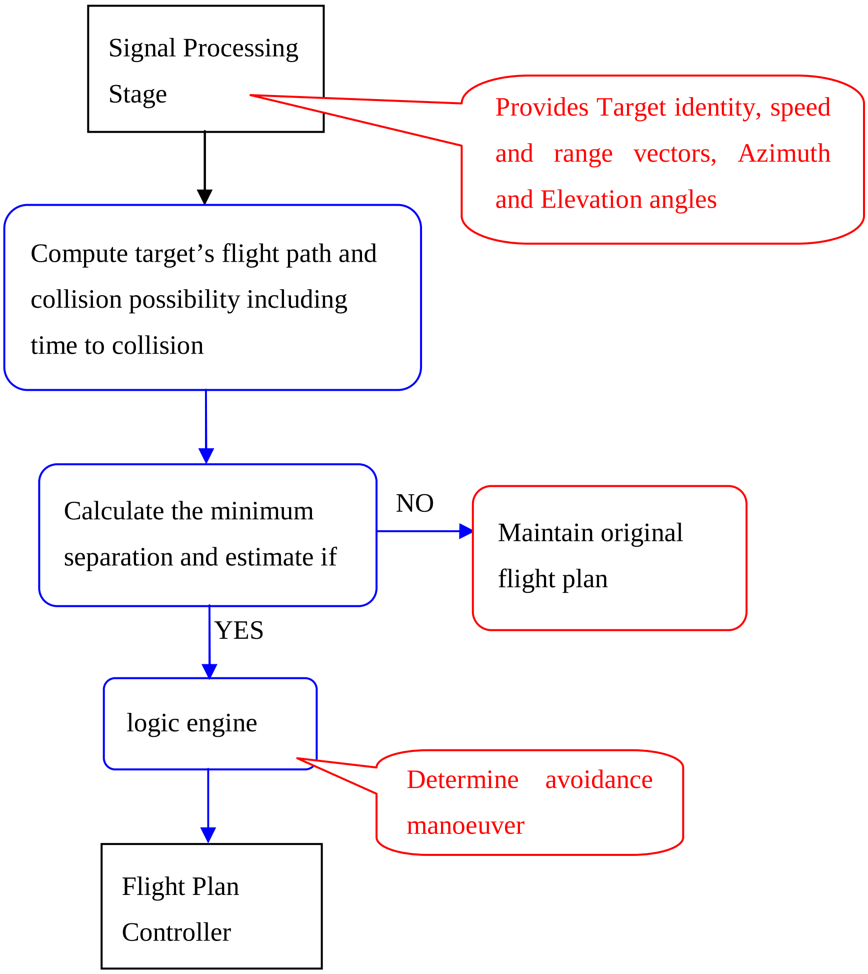

Figure 7 - Logic of collision avoidance algorithm

5.5 Changing Forward Velocity to Avoid Collision

Varying (either increasing or decreasing) the forward velocity of the Snark is the easiest and most convenient method to avoiding collisions, and it is the method that would require the least amount of change to the original flight plan.

This avoidance manoeuvre operates on the assumption that the incoming aircraft is travelling in a straight line path and at a constant velocity. Furthermore, changing the Snark's velocity to avoid a collision (whether that is an increase or decrease) will cause a transition period, as it is not an instantaneous change, but would be a gradual change.

It can be safely assumed that at speeds in excess of 15kts, the helicopter's flight controls behave more like that in a fixed-wing aircraft. [29, 30]

As explained in sections 5.1 – 5.3, the path of the incoming aircraft will be nearest to the Snark (at time t_Q) at point Q. To avoid a collision scenario, the algorithm will need to calculate a new velocity for the Snark (V_{SNEW}) that will ensure the required separation of 400m.

The mathematical derivation of this is as follows (Please refer to Figure 8 for locations, coordinates and variable names):

$$\mathrm{SP}^2 = |\overline{R_0} + \overline{V_{AS}} \times t|^2$$

$$\Rightarrow SP^2 = |\overline{R_0}|^2 + 2\overline{R_0} \bullet \overline{V_{AS}}t + |\overline{V_{AS}}|^2 \ t^2$$

$$\textit{Solving for t gives}:$$

$$t = \frac{-2\overline{R_0}\overline{V_{AS}} \pm \sqrt{(2\overline{R_0} \bullet \overline{V}_{AS})^2 - 4|\overline{R_0} - SP^2 \,||\overline{V_{AS}}|^2}}{2|\overline{V_{AS}}|^2}$$

Equation 5 - Calculation of time to reach new safe speed

In Figure 8, the Angle θ(t) represents the angle between the path of the target aircraft and the direct range vector between it and the Snark at time (t). For a collision scenario to materialise the discriminent (D) of the equation must be greater than zero when SP = 400m:

$$D = \left(2\overline{R}_0 \bullet \overline{V}_{AS}\right)^2 - 4\left|\overline{R}_0{}^2 - SP^2\right|\left|\overline{V}_{AS}\right|^2 > 0$$

Equation 6 - Relationship between determinant D and relative velocity

The reason for this is, a collision scenario between an incoming aircraft and the Snark occurs when the straight line trajectory of the incoming aircraft relative to the Snark intersects the 400m circle (in the two dimension model) around the Snark.

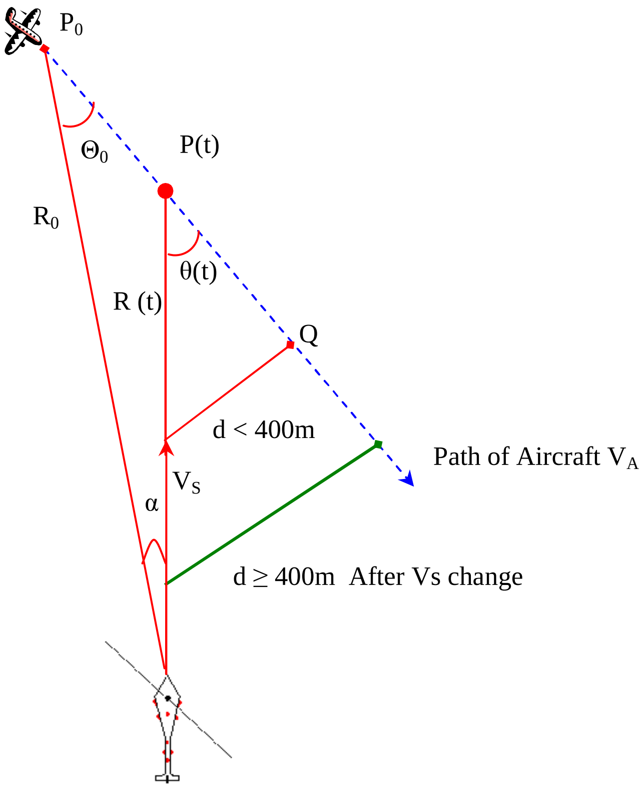

Figure 8 - Linear acceleration manoeuvre

The circle represents the imaginary 400m safe-zone surrounding the Snark, which no aircraft should intersect. If the discriminant (D) of the equation equals 0, then line is tangent to the circle, while if the discriminant (D) is a positive value, i.e. bigger than zero then the line intersects the circle at two points as shown in Figure 9. If D is negative then the line does not intersect the circle.

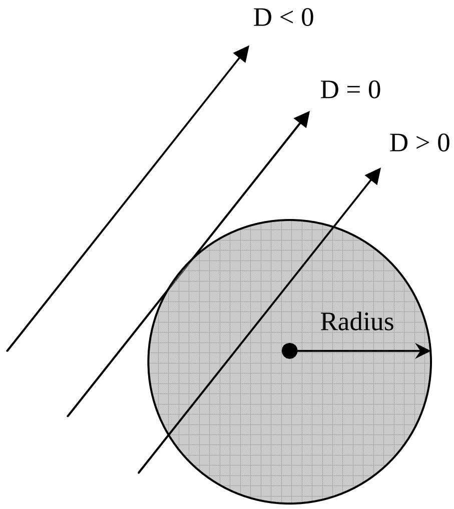

Figure 9 - Intersection of a line and a circle

To find the velocity of the Snark that can just avoid a collision or when D = 0, we specify the separation value to be 400m in the equation for D and set D equal to zero. From this we can find the time required for the Snark to reach that speed/Velocity and the time duration needed for it to travel at that speed in order to avoid collision, this is derived as follows:

$$D = 0 \Rightarrow (\overline{R}_0 \bullet \overline{V}_{AS})^2 = (\overline{R}_0{}^2 - SP^2)\times\overline{V}_{AS}{}^2$$
$$\overline{R}_0^2 \times \overline{V}_{AS}^2 \times \cos^2(\theta_0) = (\overline{R}_0{}^2 - SP^2)\times\overline{V}_{AS}^2$$

$$\Rightarrow R_0^2 \times \cos^2(\theta_0) = (R_0{}^2 - SP^2)$$

In order for the Snark to maintain a 400m gap (safety zone) between its flight path and any approaching aircraft at any time, we must ensure that the determinant in must be equal to zero.

Thus the relationship between range and separation is:

$$\cos^2(\theta_0) = \frac{(R_0{}^2 - SP^2)}{R_0{}^2}$$
$$\sin(\theta_0) = \frac{SP}{R_0}$$

Equation 7 - Separation and range as related to the offset angle

By definition, the dot product of R and V_A-V_S is:

$$\overline{R} \bullet (\overline{V}_A - \overline{V}_S) = \left|\overline{R}\right| \times \left|\overline{V}_A - \overline{V}_S\right| \times \cos(\theta_0)$$
$$R_X(V_{AX} - V_{SX}) + R_Y(V_{AY} - V_{SY}) = \sqrt{R_X^2 + R_Y^2} \times \sqrt{(V_{AX} - V_{SX})^2 + (V_{AY} - V_{SY})^2} \times \cos(\theta_0)$$
$$\because (V_{AX} - V_{SX}) = V_{ASX} \; and \; (V_{AY} - V_{SY}) = V_{ASY}$$
$$\Rightarrow R_X V_{ASX} + R_Y V_{ASY} = \sqrt{R_X^2 + R_Y^2} \times \sqrt{V_{ASX}{}^2 + V_{ASY}{}^2} \times \cos(\theta_0)$$

Squaring both sides the equation gives :

$$(R_X V_{ASX} + R_Y V_{ASY})^2 = (R_X^2 + R_Y^2)\times(V_{ASX}{}^2 + V_{ASY}{}^2)\times(1 - \frac{SP^2}{R_X{}^2 + R_Y{}^2})$$

Expanding and re-arranging the formula above we can derive a quadratic equation that relates the relative position and velocity of the incoming aircraft required to just avoid a near collision and maintain a 400m minimum safe distance.

$$(R_X V_{ASY})^2 + (R_Y V_{ASX})^2 - 2(R_X V_{ASX} R_Y V_{ASY}) - RB^2(V_{ASX}{}^2 + V_{ASY}{}^2) = 0$$

Equation 8 - Relationship between relative aircraft position to relative aircraft speed at minimum 400m separation (RB)

To calculate the forward velocity of the Snark required to avoid the collision we solve the equation in V_{ASY} as the Snark is assumed to maintain its current heading, $V_{SX} = 0$, and V_{ASX} is not changed by this manoeuvre.

Rearranging, gives:

$$\therefore (R_X{}^2 - RB^2) \times V_{ASY}{}^2 - 2(R_X V_{ASX} R_Y) \times V_{ASY} + V_{ASX}{}^2 (R_Y{}^2 - RB^2) = 0$$

From the above, we can estimate the new forward velocity for the Snark needed to ensure a 400m-separation distance (i.e. aircraft is tangent to sphere) by solving the above equation for V_{ASY}, we get:

$$V_{ASY} = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A} \Rightarrow V_{SY\,NEW} = V_A - \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$
$$\text{Where}$$
$$A = (R_X{}^2 - RB^2)$$
$$B = -2(R_X V_{ASX} R_Y)$$
$$C = (R_Y{}^2 - RB^2) V_{ASX}^2$$

Equation 9 - Calculation of velocity to avoid collision (2D)

Equation 9 gives two values for the forward velocity of the Snark that will set the minimum separation to 400m and avoid the collision. Collision can be avoided if the velocity of the Snark is either increased or decreased (as Equation 9 yields two values, a high and low possible speed values).

Which of these values will be selected depending on whether the Snark can accelerate or decelerate in time to reach that velocity, or if the velocity values are within the Snark's operational capability.

We can expand the universal quadratic formula to include an altitude consideration, i.e. converting it to a 3D case:

$$(R_X V_{ASY})^2 + (R_X V_{ASZ})^2 + (R_Y V_{ASX})^2 + (R_Y V_{ASZ})^2 + (R_Z V_{ASX})^2 + (R_Z V_{ASY})^2$$
$$- 2(R_X V_{ASX} R_Y V_{ASY}) - 2(R_X V_{ASX} R_Z V_{ASZ}) - 2(R_Y V_{ASY} R_Z V_{ASZ})$$
$$- RB^2 (V_{ASX}{}^2 + V_{ASY}{}^2 + V_{ASZ}{}^2) = 0$$

Equation 10 - Relationship between relative aircraft position to relative aircraft speed at minimum separation (including altitude and climb differences)

Again solving the above equation for V_{ASY} gives:

$$V_{ASY} = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A} \Rightarrow V_{SY\ NEW} = V_{AY} - \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$

with

$$A = (R_X{}^2 + R_Z{}^2 - RB^2)$$

$$B = -2(R_X R_Y V_{ASX} + R_Y R_Z V_{ASZ})$$

$$C = (R_Y{}^2 + R_Z{}^2 - RB^2)V_{ASX}^2 + (R_X{}^2 + R_Y{}^2 - RB^2)V_{ASZ}^2 - 2R_X R_Z V_{ASX} V_{ASZ}$$

Equation 11 - Calculation of new collision avoidance velocity (3D version)

There will be some cases where an exclusive speed variation manoeuvre will not be sufficient to avoid a collision, and a turning manoeuvre or a change in altitude will be required.

Another solution would be for the Snark to reach a hover stage until it is possible to resume its path. The hovering action is an extension of a slowing down to a stop action.

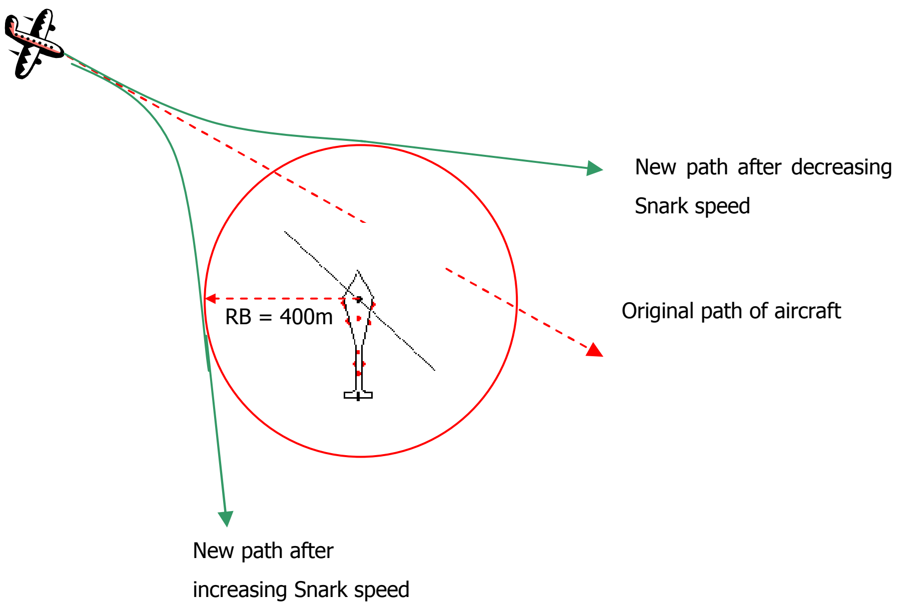

Figure 10 - Collision avoidance by changing to either a lower or a higher travel velocity

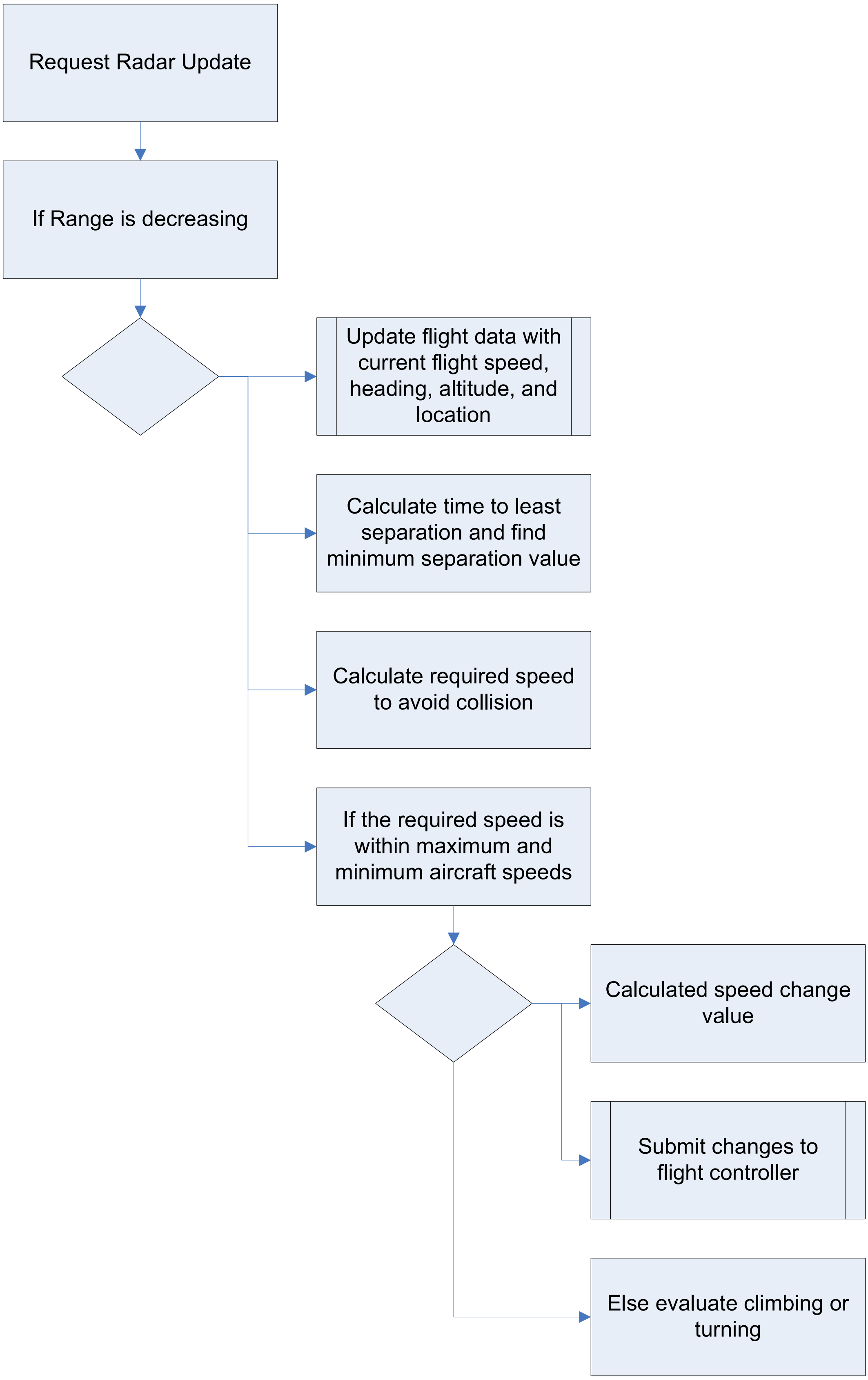

Figure 11 - Block diagram for adjusting speed program

5.6 Changing Altitude

Another method for avoiding a collision is by changing the altitude of the flight path.

By re-forming, Equation 8 in Section 5.5 we can calculate the 3D velocity at which the Snark would be on a safe path with 400m minimum separation

$$(R_X V_{ASY})^2 + (R_X V_{ASZ})^2 + (R_Y V_{ASX})^2 + (R_Y V_{ASZ})^2 + (R_Z V_{ASX})^2 + (R_Z V_{ASY})^2$$
$$-2(R_X V_{ASX} R_Y V_{ASY}) - 2(R_X V_{ASX} R_Z V_{ASZ}) - 2(R_Y V_{ASY} R_Z V_{ASZ})$$
$$-RB^2(V_{ASX}{}^2 + V_{ASY}{}^2 + V_{ASZ}{}^2) = 0$$

Re-arranging the equation and solving the term for R_Z (the separation of flight levels between the two aircraft), we can obtain a target altitude for the flight path to avoid a collision.

$$R_Z = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A} \Rightarrow \text{New vertical separation} = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$

where

$$A = V_{ASX}^2 + V_{ASY}^2$$
$$B = -2(R_X V_{ASX} V_{ASZ} - R_Y V_{ASY} V_{ASZ})$$
$$C = R_X{}^2(V_{ASY}^2 + V_{ASZ}^2) + R_Y{}^2(V_{ASX}^2 + V_{ASZ}^2) - RB^2(V_{ASX}^2 + V_{ASY}^2 + V_{ASZ}^2) - 2(R_X V_{ASX} R_X V_{ASY})$$
$$and\ RB = 400m$$

Equation 12 - Safe altitude calculation

Once that flight level is established, we introduce a vertical lift (or drop) in the direction of flight this is achieved in the program through the introduction of a vertical component of the velocity ($V_{S.Z}$) to the component speed.

5.7 Changing the Heading

Performing a change in flight direction of the Snark is perhaps the most exhaustive avoidance manoeuvre that can be performed during flight, especially in terms of fuel consumption, deviation from flight path and mission completion time. However, changing flight path or direction provides the only chance of resolving head on collisions and some other scenarios where changing the speed (but not flight direction) will not result in a safe solution (or when the speeds required are too great).

The mathematical analysis of a turning manoeuvre can be seen in Figure 12, where the Snark has to undertake two turning actions in order to avoid a single incoming target, passing through key way points. The first turn is to change its heading away from the incoming aircraft and the second to resume its initial heading once a safe separation distance has been achieved.

One of the conditions requires turning manoeuvre is that the action of changing only the forward speed of the Snark will not ensure a safe flight path, hence it is deemed necessary to change the direction of the Snark. As before, the Snark must identify the value of the minimum distance between the target airplane and its flight path at the point of intersection (distance d).

Using Equation 4 listed earlier for the calculation of separation distance, we calculate the initial separation between the two aircraft. If the separation value is below the 400m safe-limit, and a forward velocity change cannot avoid a collision we initiate changing the direction of the aircraft. When initiating a turning manoeuvre, the timing, distance and instantaneous locations of both aircraft must be compared at all times to ensure a safe and accurate manoeuvre with minimal adjustment. [31]

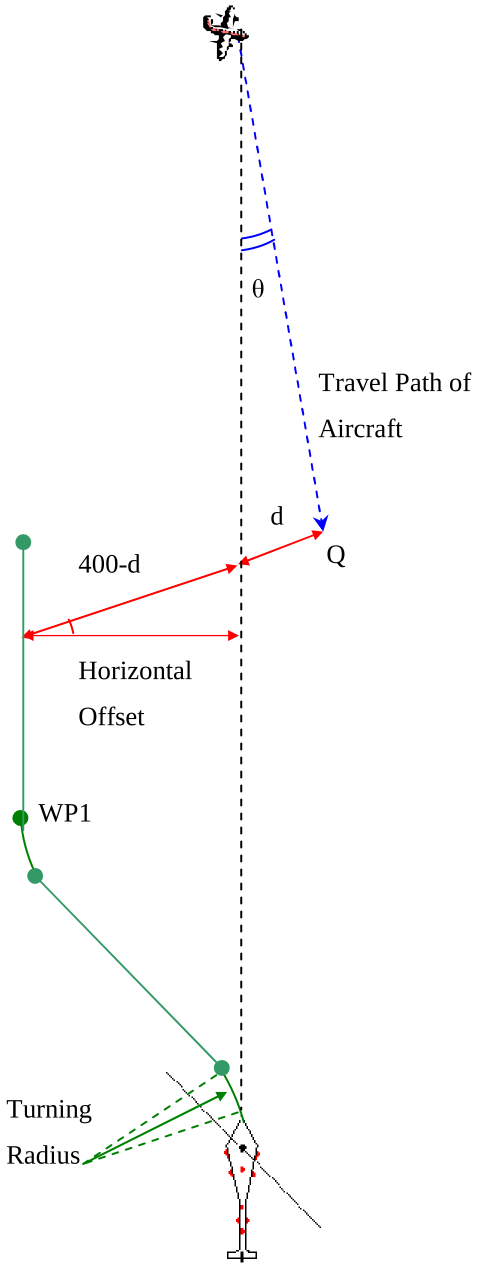

Figure 12 - Turning manoeuvre

The actual turning or changing direction of the aircraft is the same, and it is carried after the calculation of the separation distance (d) and the command is issued to start turning, the collision avoidance algorithm would need to estimate the turning radius, the rate of turn (ROT), banking angle and velocity.

The banking angle and the radius of turn together determine the rate at which the aircraft can change its direction and choosing these two values carefully can result in optimum turn performance and reduced stress on the airframe. [40]

Prouty [29] argued that the basic aerodynamic behaviour of a turning helicopter could be expressed as exactly that of a conventional fixed wing aircraft, as the principle of generating lift is the same on both fixed-wings and rotating blades. However, the available power from the engine of the helicopters (in a turning manoeuvre) degrades dramatically with the banking angle and load factor (G-Load) of that manoeuvre.

In order for the helicopter to maintain its altitude and speed at a turn (with a banking angle of 60°), the rotor/engine must develop twice as much thrust as in straight and level flight. [29] and [30]

Therefore, to achieve optimum turn performance, and to limit structural stresses; we propose to use banking angles between 18º and the optimal 35º off the vertical axis of the rotor.

It was decided that the recommended banking angle throughout the avoidance manoeuvre should be approximately 25 º off the vertical axis of the rotor shaft this corresponds to a thrust/engine load factor of 1.066G. [40]

Furthermore, NASA’s ERAST study program has shown that the primary factor in displacing an aircraft is the bank angle. For a given bank angle, the time required to displace the Snark by approximately 500 feet is independent of the aircraft’s velocity [35-37].

The turning manoeuvre can be calculated as follows:

Aerodynamic simulation of a turning aircraft yields:

$$R = \frac{V^2}{g \times \tan(\phi)}$$

Where

R is the radius of curvature of the turn

V is the velocity

g is the acceleration

φ is the banking angle in Radians

Further more: the distance covered by the aircraft during this manoeuvre is equal to the length of the curve L

$$L = R\phi$$

And the time to perform the turn is:

$$t = \frac{L}{V}$$

The Rate of Turn (ROT) is

$$\frac{d\phi}{dt} = \frac{\phi}{T} = \frac{\phi}{L/V} = \frac{\phi V}{L} = \frac{\phi V}{R\phi} = \frac{V}{R}$$

$$\Rightarrow \frac{V/V^2}{g \tan\phi} = \frac{g \tan\phi}{V}$$

$$\therefore ROT = \frac{g \tan\phi}{V}$$

To estimate the time to reach the midway point (WP1) or the time around the curve, we can derive the following formula:

$$\text{The distance to Midway Point (WP1) is}$$

$$\text{Midway} = \text{Radius} - \text{Radius} \times \text{Cos}(\phi \times \text{CurveTime})$$

$$\text{Midway} = \text{Radius} \times (1 - \text{Cos}(\phi \times \text{CurveTime}))$$

$$\frac{\text{Midway}}{\text{Radius}} = 1 - \text{Cos}(\phi \times \text{CurveTime})$$

$$\text{Cos}(\phi \times \text{CurveTime}) = 1 - \frac{\text{Midway}}{\text{Radius}}$$

$$\therefore \text{CurveTime} = \frac{1}{\phi} \times \text{Cos}^{-1}(1 - \frac{\text{Midway}}{\text{Radius}})$$

From this derivation, we can calculate the ROT for the Snark and the turning radius in the simulation program according to the following:

$$\text{ROT} = \frac{g \times \tan(\phi)}{V} \qquad \text{Turning Radius} = \frac{V^2}{g \times \tan(\phi)}$$

$$\text{CurveTime} = \frac{1}{\phi} \times \cos^{-1}(1 - \frac{\text{Midway}}{\text{Radius}})$$

Equation 13 - Calculation of turning radius, rate of turn (ROT) and turn time

Having estimated the values for the rate of turn and turning radius, the collision avoidance algorithm program can now calculate the values of the length of the curve segment and the time that the Snark requires to complete that turn (i.e. the time needed to reach the safe point).

However, in maintaining the realistic approach to this, and from the information gathered concerning the flight control module used in the aircraft, the algorithm would most likely compare the current location of the Snark during the manoeuvre to the defined safe waypoint offset to its path.

In the 3D environment, where one can account for any difference in altitude and include it in the total separation between the two aircraft and we can estimate the horizontal offset distance for follows:

$$\text{Horizontal travel} = \sqrt{(400^2 - dy^2 - dz^2)}$$
$$\text{Midway Point} = (\text{Horizontal offset} - \text{original } dx)$$

Equation 14 - Calculation of horizontal travel distance and midway turning point

Through this approach, the Snark would compare its location (GPS location in space) with the intent or waypoint (WP) location.

In the instant the command is given to turn to a particular direction, the Snark would follow part of a circular path (around the curve drawn by the radius of turn and the angle of turn), at a constant ROT (calculated above). Until it reaches the midway point, at which the Snark would reverse its rate of turn in order to return to the initial Snark heading.

Standards flight rules require that if the trajectories of two aircraft travelling in opposite directions approach one another at distances less than 150ft, then both aircraft must turn right. [25]

However, the operational mode of the Snark during flight would most certainly be the AUTO flight mode, whereby a waypoint would be provided to the flight computer and it would automatically select the heading, banking angle, turning radius, and the speed while turning.

This forces the Snark to adapt a right turn manoeuvre, even though the target could be more efficiently avoided if the Snark turns to the left instead. This is because the algorithm is designed to monitor the Snark surroundings at all time and does not commit the Snark to a rigid turning scheme where the safety situation might change suddenly.

The algorithm would provide the flight controller with the following information:

- Heading of waypoint as calculated
- Time around the curved path to reach the safe distance
- X, Y, & Z- coordinates of the waypoints (WP1-5 as previous figure)

The proposed method for this is that the collision avoidance algorithm would transmit the location of the next way point to the flight controller module as the Snark is en-route to it. E.g. on the instant of detecting a collision scenario, the Snark would start to change its path (turn) away from target, it would transmit the location (X, Y, & Z-coordinates) of WP1, in addition to the ROT, radius, manoeuvre time etc.

Before it reaches WP1; the flight controller then changes the heading of the Snark and commits the Snark to returning to its original heading albeit at a parallel path to its original path.

5.8 Avoiding Multiple Aircraft

In certain cases, the Snark may have to avoid more than one aircraft. In circumstances such as flying near an airport landing strip, or in semi-regulated civilian airspace, the collision avoidance algorithm must be able to cope with such situations.

Although the manoeuvres and counter actions to ensure a safe path for the Snark would be the same as for avoiding a single aircraft, it is necessary that these measures do not put the Snark in a collision path with any other aircraft.

In addition, the Snark would deviate considerably from its original flight path if it avoided each collision scenario as a separate case, i.e. changed its course each time as it detects a collision possibility with any incoming aircraft individually rather than using more collective approach, the large deviation could cause considerable losses in both fuel and mission time.

Such a collective approach is known as strategic real-Time path optimization as opposed to the tactical real-time path optimization approach when dealing with individual collision scenarios [9].

One method indicated by Karhoff *et al* [41] is to determine the most threatening aircraft, then determine which of these are with in 3 mile/5km, and calculate a safe flight path to avoid three of them.

However, this method was assumed for fast UAV aircraft with a considerable advantage in speed over the avoided targets, rather than an unmanned helicopter (namely the Snark) flying at a speed disadvantage. Therefore we modify this technique to suit the Snark’s slower speeds, turning performance, and response to change of path command.

Another method was outlined by the US Navy during their experimenting on unmanned surface vehicles (USVs). The space and naval warfare systems center in San Diego is developing core technologies required for robust USV operation in a real-world environment, primarily focusing on autonomous navigation, obstacle avoidance, and path planning. [42]

The recommended approach is to use spatial simulation to establish a no go area, which the target would pass through at a specific time.

Once the algorithm calculates a safe point for target A, The avoidance program would calculate the collision possibility with other aircraft at that point, by using the projected location, heading, and speed of the Snark while traveling to that point and when reaching that point.

Although the working of this method is very similar to our proposed idea, however the logic of this method becomes too complicated for use on a 4D (x,y,z, time) when applied in aerial avoidance., due to the altitude factor and the possibility of targets in a climbing or descending trajectory.

Furthermore, the original method requires dividing the path area into sectors (squares) such to define the avoidance areas, this can be very complicated and a time costly exercise to do in a 3D environment for aircraft with limited computing capabilities. [42]

Therefore, we propose to use a different method where in order to avoid multiple aircraft, the collision avoidance algorithm will investigate the possibility of flying to a designated safe point, whereby reaching that safe point would eliminate all the dangers of the approaching aircraft.

This approach would require the Snark to fly under the FLYTO flight mode, where it would automatically determine the type of manoeuvre or the required change to the current flight parameters (direction, airspeed, altitude, and heading) in order to pass through that way point at the required time. This can be made simpler by utilizing the flight controller's autopilot ability to navigate to predetermined way points or path points.

This algorithm is based upon avoiding the paths of the incoming aircraft at all times rather than a time based avoidance algorithm that avoids the incoming aircraft only at the critical time of closest approach. This would simplify the calculation of a single waypoint, by simply converting the paths of the other aircraft into no-go/keep clear zones that must be avoided at all times.

Once the mathematical model for this conversion is obtained, a single safe waypoint can be calculated by calculating the coordinates of a waypoint that is 400m (or more) away from the paths of all detected aircraft. Furthermore, if two or more of these aircraft have intersecting paths, then the algorithm should be able to find a safe waypoint from the point of intersection (refer to Figure 13)

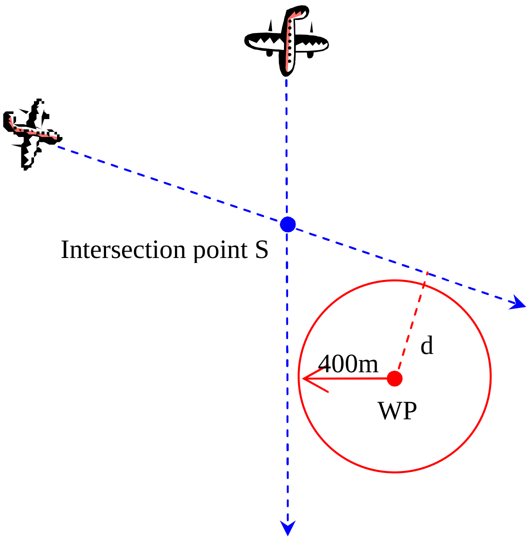

Figure 13 - Avoiding two intersecting aircraft flight paths

As can be seen from the figure above, the Snark can find a safe point at WP once the intersection point between the flight paths of aircraft 1 and 2 is calculated.

We caluclate the coordinates of intersection point S, as followes:

Assuming that the incoming aircraft maintains a straight ling flight path, then the path of aircraft (N) can be calculated as

$$\overline{P_N(t)} = \overline{P_N(0)} + \overline{V_N} \times t$$

Where $\overline{P_N(t)}$ is the location of the aircraft at time t, $\overline{P_N(0)}$ is the initial location of the aircraft, and $\overline{V_N}$ is the velocity vector of the aircraft.

However, to simplify the calculation and, as described earlier, to eliminate time as a factor in the avoidance manoeuvre (stay clear of the flight path of all aircraft at all times) we must eliminate the time (t) term in the equation.

Suppose the flight paths of the two incoming aircraft intersect as point S. Note that the two aircraft will not both at this point at the same time.

Then for each aircraft there is a different time (t), where

$$\overline{S} = \overline{P_N(0)} + \overline{V_N} \times t$$

Therefore, for each aircraft:

$$S_X = P_{X_N} + V_{X_N} t_N$$
$$S_Y = P_{Y_N} + V_{Y_N} t_N$$

Where P_{XN} and P_{YN} are the components of $P_{N(0)}$

Eliminating T_N

$$S_X V_{Y_N} - \backslash S_Y V_{X_N} = P_{X_N} V_{Y_N} - P_Y V_{X_N}$$

Therefore, for the two aircraft we get:

$$S_X V_{Y1} - S_Y V_{X1} = P_{X1} V_{Y1} - P_{Y1} V_{X1}$$
$$S_X V_{Y2} - S_Y V_{X2} = P_{X2} V_{Y2} - P_{Y2} V_{X2}$$

Solving the above simultaneous equations for S_X, we get the following:

$$S_X = \frac{(P_{X1}V_{Y1} - P_{Y1}V_{X1})V_{X2} - (P_{X2}V_{Y2} - P_{Y2}V_{X2})V_{X1}}{V_{X2}V_{Y1} - V_{Y2}V_{X1}}$$

And :

$$S_Y = \frac{(P_{X1}V_{Y1} - P_{Y1}V_{X1})V_{Y2} - (P_{X2}V_{Y2} - P_{Y2}V_{X2})V_{Y1}}{V_{X2}V_{Y1} - V_{Y2}V_{X1}}$$

Once the relative coordinates of the intersection point are identified, calculating the coordinates of four possible safe waypoints surrounding the intersection point can be easily achieved (refer to Figure 14).

The centre of the waypoint occurs along the imaginary line that bisects the angle formed between the two paths of the aircraft. Therefore a right angle triangle is formed between the distance L along the bisecting line, the 400m radius perpendicular to the path of the aircraft, and the flight path of the incoming aircraft.

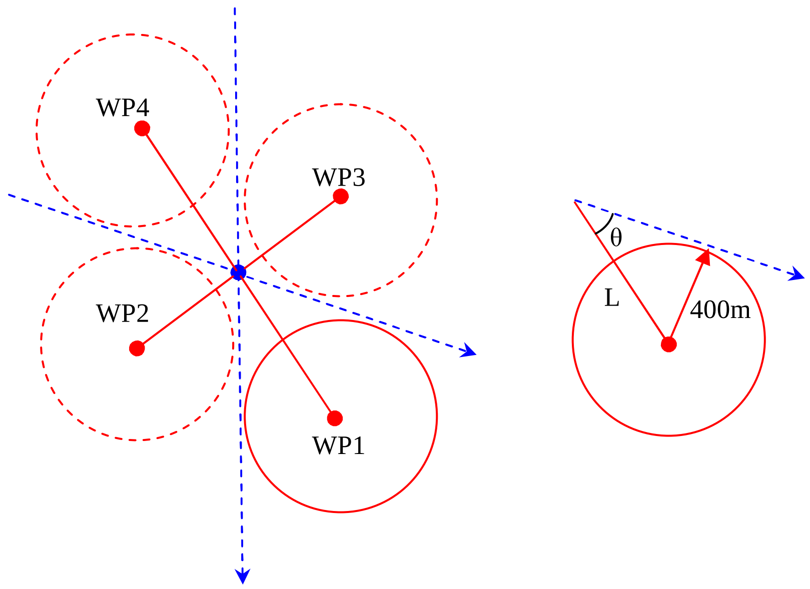

Figure 14 - Possible waypoint locations and distance along bisecting line L

The distance (L) along the safe line can be calculated as follows: $L = \frac{400m}{Sin(\theta)}$

Once the distance L is calculated, the location of the waypoint can be easily obtained by adding the coordinates of the intersection point to the coordinates of the point at the end of line L.

Once all of the four waypoints are identified the algorithm then calculates:

- The time required for the Snark to reach each of these waypoints
- The distance to that waypoint
- If waypoint is also safe for other aircraft

It is quite probable that the algorithm will find several waypoints that will be suitable for avoiding of all of the flight paths of the incoming aircraft. In this case, the algorithm will select one of these waypoints to fly to, depending on the distance to the waypoint, time to reach that waypoint, and the change in velocity. In this example point WP1 indicated with the solid line safe zone represents the best option for the Snark.

If the paths of two incoming aircraft are parallel or do not intersect (i.e. the two aircraft are diverging), the algorithm wills use the turning avoidance process calculation to find a way point to the left or to the right of these two aircraft.

The algorithm will evaluate each waypoint according to the following criteria:

- The time to reach that waypoint must be equal to or less than the minimum available avoidance time
- The distance to that waypoint (d_{WP}) is the least amongst all other waypoints (if multiple safe points are calculated)
- The velocity required to reach that waypoint must be within the Snark’s capability
- The waypoint should lie at least 400m away from the flight paths of all incoming aircraft detected thus far and that it does not lie in the direct path of any aircraft.
- The algorithm selects the waypoint that satisfies all of the above criteria and requires the least amount of change in Snark’s airspeed, and the waypoint that takes the least time to reach.

In the example shown above, the Snark would most likely choose to fly to point WP1 (if the flight paths and velocities of the targets remain constant) as it is the closest point to the Snark’s original location, and it requires the least change in velocity and heading in order to be reached within time.

This approach can be easily adapted to any number of aircraft, where the algorithm would find four safe waypoint for each pair of aircraft, starting with aircraft 1 through aircraft n, where it would estimate if the two aircraft do intersect, and if they are then it would a suitable waypoint, that is nearer to the Snark than all of the others, and requires the least amount of velocity change to reach.

The number of safe waypoints is estimated at:

$\text{number of waypoints} = 2 \times n(n-1)$, where n is the number of aircraft approaching the Snark, i.e. if three aircraft approach the Snark at various angles, then the algorithm can identify 12 safe waypoints that can be flown to.

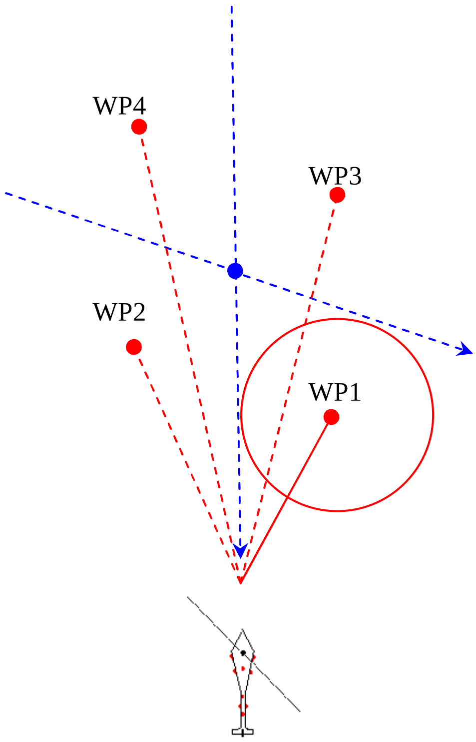

Figure 15 - Selecting closest waypoint

CHAPTER 6

6.1 ARCHITECTURE OF SIMULATION PROGRAM

This chapter describes how the software is structured, including how the Snark is simulated and how it interfaces with the collision avoidance algorithm with other parts/functions of the flight control unit.

The program is comprised of several sub-programs or functions; each function performs a specific task within a specified time frame or instant. The reason for this approach is to ensure that only the main function controls and modifies any variable specific to that aircraft (for example range, attitude, elevation and azimuth angles, time to collision etc). This would ensure a modular approach to the design of the program and a great deal of flexibility when calling and initiating all of the sub-functions. It will also increase the system's error handling capabilities.

The complete program structure comprises of the following sub-programs or functions:

Main function: The main function in this program acts as an embedded sub-routine that interfaces to the main flight controller code. This sub-routine continuously monitors information from the radar unit, and evaluates if any of the detected aircraft are approaching the Snark. If one or more of those targets is approaching the Snark (i.e. if the range decreases), it will pass on the flight parameters (range, relative airspeed, heading, and detection angles) to the avoidance function. The main function can be used to read GPS data input and identify if the Snark is on the correct flight plan.

Radar module: The radar module in this program is designed to simulate the operation of the radar called from the main function. The output from the radar function comprises of the following for each detected aircraft and always represents the current values of the aircraft's data:

- Target range or distance from Snark at time t (R)
- Azimuth angle or horizontal displacement angle from Snark heading (α)
- Elevation angle or vertical displacement angle from Snark heading (β)
- Simulated GPS coordinates to indicate the target's location in space (TR)
- Simulated GPS coordinates to indicated the Snark's location in space (SLOC)
- The radar function according to the performance of the MiSAR radar system. [43]

Avoidance module: This source file contains the definitions and methods of the functions used by the collision avoidance algorithm. The output of the avoidance function comprises of:

- Forward airspeed/acceleration, new altitude/climb rate, and rate of turn if the system is avoiding one aircraft operating under the MANUAL & DIRECTED flight control settings.

- Geospatial location of a safe way point that the Snark can fly to avoid a collision, if the system is avoiding multiple aircraft and is restricted to flying under FLYTO mode. The waypoint data structure contains the coordinates of the waypoint relative to the initial position and heading of the Snark, and the time and velocity required to reach that waypoint.

Flight controller module: This module contains a representation of the flight controller used by the Snark, and it contains data structures of the current and original headings and flight paths of the Snark. The function also contains the necessary path change algorithms and equations, once the avoidance function determines that there exists a need to change the heading to avoid a collision. Where the flight controller would determine which path to take, rate of turn, direction, and in turn calculate the necessary changes to current flight speed in order to reach the waypoint within the specified time frame.

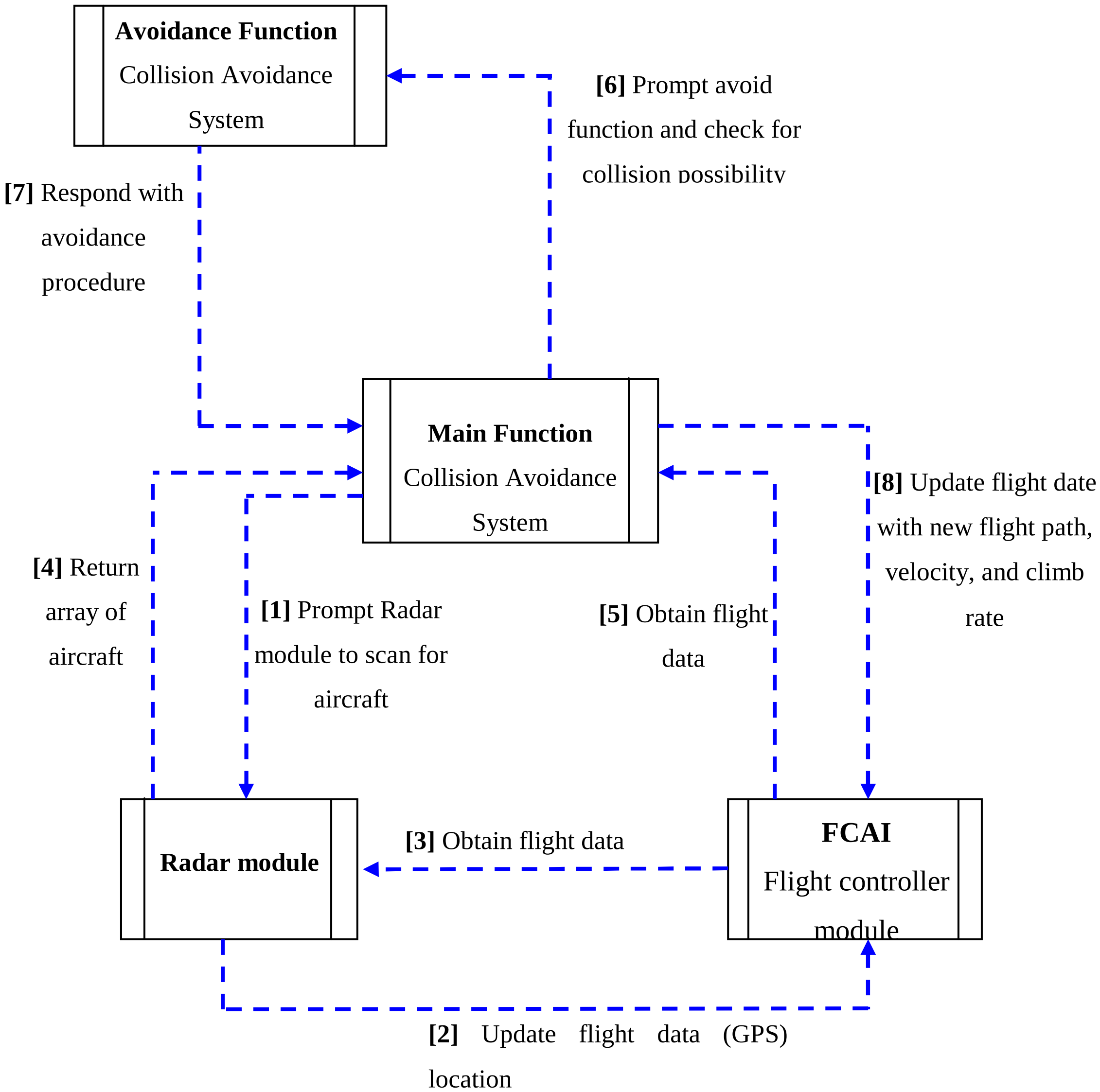

Figure 16 - Block process for subsystem integration

Figure 16 shows how the modules operate as part of the complete flight path system. Here the radar module of the project would be responsible for controlling and correlating the target data from antennas located around the aircraft and processes the raw data into streams of information containing each detected target's range and velocity vectors.

This information is read by the main function and passed to the avoidance algorithm which identifies possible collisions and calculates a suitable avoidance action or actions. The main function then conveys the avoidance information to the flight control system.

6.2 Operation Sequence

Due to the experimental nature of this project and the lack of exact information concerning the radar unit and flight controller module it is assumed that the communication between the three units is achieved via data structure or variable exchange.

The operational sequence of the simulation program is as follows:

- The avoidance algorithm/main program would "ping" or request information from the radar unit
- The radar unit would respond by providing a list of the target located in the scan zone, in the form of an array containing structures. Each of these structures contains an ID for the target, range, elevation and azimuth angles and perhaps velocity of the target, if the radar unit contains an advanced post-scan data processing capability. Another option would be that the radar unit would provide a continuous scan option for the targets according to a preset interval.
- The avoidance algorithm requests an update from the flight controller concerning the Snark's current heading, airspeed, mode of operation (FLYTO, MANUAL, OR DIRECTED), next way point and the distance to that way point.
- The avoidance algorithm would assess the possible collision scenario, estimate if there were any collision or any infringement to the 400m-clearance bubble, and would provide the most suitable action to take.
- The avoid algorithm would instruct the flight controller to change the flight path of the Snark in the case of an imminent collision. This would take the form of a packet of data containing the following information: coordinates of the new waypoint, distance to that waypoint from current locations, time to reach that waypoint, new heading, new airspeed, and rate of turn, time to perform manoeuvre, and the coordinates of way points.
- The flight controller would instigate these changes and would update the heading for the radar system.

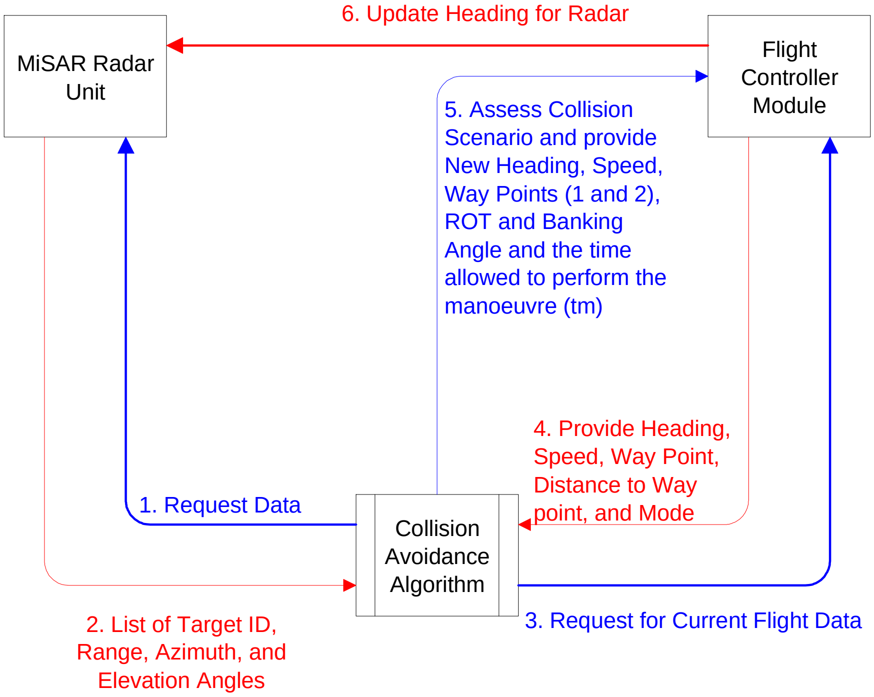

Figure 17 - Interfacing exchange diagram

6.3 Selection of Avoidance Manoeuvre

Once the program (namely the separation loss algorithm) has detected a collision with a single aircraft, and no other aircraft exist in the vicinity of the radar scan, the program applies the following process:

If the number of detected aircraft is just 1, and that aircraft is on a collision course, the avoid function calculates the least separation and the time at which this least separation distance occurs. After obtaining those values, the avoidance function algorithm compares the three avoidance methods (as discussed in chapter 5) as to which is the most suitable to avoid this particular incident.

When the algorithm has identified which of the three different manoeuvres is most appropriate (changing speed, or altitude, or turning/changing heading), then the avoidance function initiates that manoeuvre, and instructs the flight controller module with the necessary changes to speed, heading, and climb rates. Once the danger has passed, the avoid function algorithms would become inactive again, and would relinquish control back to the original flight controller flight plan.

In the case of avoiding multiple targets, the avoidance algorithm would add new safe waypoints to the original flight plan (itself a series of waypoints).

6.4 Types and Structure of Exchanged Data Entities

To further expand on the discussion from the previous section, we describe several types of structures that are used for the specific task for each function.

6.4.1 Data from Radar Unit to Main Function

The information exchange with the radar unit consists of acquiring:

- ID: a serial number or identification code unique to each detected target
- Range for that target
- Azimuth angle (α): In conjunction with the range, this indicates the location of the target in the X and Y coordinates as relative to the Snark
- Elevation angle (β): In conjunction with range, this indicates the altitude of the target aircraft, as compared to that of the Snark
- Relative velocity of aircraft
- Relative heading of aircraft
- Range to aircraft
- Time to collision of that aircraft

The above information is in the form of a data packet (more specifically a C structure) for each target and is passed to the Main function/Collision Avoidance function as an array containing the details for each individual target.

6.4.2 Data from Main Program to Collision Avoidance

The main/collision avoidance function comprises of several functions, primarily the collision assessment and the collision avoidance functions, thus it is best to describe the data variables needed by both functions:

The collision assessment function is where the location and heading of the detected aircraft is analysed for assessing collision scenarios; the input variables needed for this stage are obtained from the radar unit and they are:

- Target ID
- Range at current time relevant to the Snark
- Azimuth attitude angle at current time relevant to the Snark heading (α)
- Elevation attitude angle at current time relevant to the Snark (β)

In addition, the collision avoidance function would request current flight data from the flight controller in the aircraft, as to determine the Snark's current heading and altitude information:

- Current heading
- Current Snark velocity
- Altitude
- Next waypoint
- Time to next waypoint
- Distance to waypoint in relation to the Snark's current location

These data entities are used to determine the best action to avoid a collision scenario

6.4.3 Data from Collision Avoidance Function to Flight Controller

The collision avoidance function, analyses the information collected earlier (by the assessment part) and generates the most appropriate action to avoid a collision by changing the flight characteristics of the Snark by transmitting a data packet to the flight controller containing the following information:

- New heading at the next waypoint (Aircraft Heading)
- New velocity for the Snark to travel at to the next waypoint (V_{WP}) and necessary velocity change (V_{CHANGE}) as will be calculated by the Avoidance Algorithm
- Coordinates of the safe way point (x, y, altitude)
- Time to reach that waypoint from current location (t_{WP})
- Distance to waypoint from current positions (d_{WP})

CHAPTER 7

TESTING PROCEDURE

7.1 Testing Objectives

In order for the system to function reliably under all circumstances, a properly established testing procedure was followed. This testing procedure included several separate stages of evaluation throughout the project's life span. The testing procedure was arranged to determine the different factors that affect the performance of the system.

These included:

- System accuracy and detection time
- Tracking ability of the algorithm
- System reliability and manoeuvre response time and accuracy of distance and separation.
- System efficiency and most importantly the reliability of the programme
- Fault/error conditions: i.e. what are the parameters that would cause the system to halt/crash or would become inaccurate (example increasing the number of targets, speeds, variations in directions, etc)
- Time of system response
- Adaptability of the system
- System limitations as to the maximum number of aircraft that can be avoided safely

The tests conducted through out this experiment have been designed to investigate the ability of the program/algorithm to sense, assess, and avoid a possible collision with another aircraft under normal flight circumstances.

As discussed in earlier chapters, the collision avoidance response of the program is based on the strict CFR 91.113 flight rules. Since the algorithm is designed for the specific purpose of avoiding civilian aircraft in sanctioned controlled airspace, it is assumed that the test conditions and collision scenarios should be based on these safety standards.

The test scenarios and procedures conducted throughout this experiment are loosely based upon the published experiments conducted by NASA during their sense and avoid ERAST project [12] & [44]. However, the tests were changed to reflect the difference in flight performance between their test aircraft and our helicopter based platform. The differences in speed, climb, and manoeuvring capability influence the decision making process for the algorithm and the eventual way point path sequence or action undertaken. Also, tests scenarios evaluated by Grilley [23] and the report notes reported by

Ebdon [45] were also incorporated into the testing watch lists, specifically in head-on collisions and overtaking scenarios.

7.2 Testing Procedure/Process

From the various discussions in the previous chapters, the testing procedure includes the following scenarios:

- Single target detection scenario
- Static and moving target
- Constant speed/direction
- Targets flying in intersecting or parallel courses
- Multiple target detection scenarios

Initially, the program was tested using a 2-D environment, where all incoming aircraft are at the same altitude.

The second phase is to convert the program to evaluate the 3-D environment to further assess a real life situation.

The simulation program examined scenarios that might be encountered by the Snark as follows:

- Program input was the target's initial range, and azimuth angle relative to the Snark
- Target speed and heading were input to the program as well, however they were used by a separate function (Target Path Estimator) to estimate the target flight path rather than being used by the detection/avoidance algorithm directly
- Target path estimation would act as the input to the simulated radar system, where it provides the detection/tracking function with up to date values of range (R) and azimuth (α angle).
- The main program loop evaluates the varying ranges and consequently estimates if the tracked target is on a possible collision course. For example it will monitor if the range to the target is getting smaller with every update and if the calculated separation at closest approach point is below the minimum safety distance of 400m.
- In the case of a possible collision scenario, the main program would call the collision estimation function, which would calculate the speed of the target; the time to collision, and minimum separation. It would also act as a final check to verify the possibility of a collision or an intrusion in the 400m safe-zone.
- If a collision is predicted, the program would initiate the collision-avoidance logic engine, to find the best collision avoidance scheme.

In the initial stages of the project, the MATLAB simulation environment was used to identify and plot the movement of the Snark and the approaching aircraft, as it was easier to visually identify the movements and optimize the algorithm.

Initially, a mock system was designed on MATLAB, that included a simple program for testing the detection algorithm. This was expanded to include a code section that simulated the logic model statements.

This testing stage was simulated on a desktop computer equipped with MATLAB, and did not take place on the actual single board computer that would be used in the final product.

The next stage was to transform the MATLAB simulation code into C language code, to create the actual C files that were to be incorporated in the actual aircraft software.

As a preparation for the testing, it was only logical to divide the testing process into the following sections:

Due to the difference in procedure and the evasion manoeuvre response, it is best to divide the testing process into:

- Single target avoidance: where we would test basic avoidance techniques, not only ensuring adequate separation, but also the method and manoeuvring commands: change of speed, turning, or climbing for instance. This method can be particularly useful if the UAV is under the control of a ground operator or MANUAL mode as the real time avoidance commands would alert the ground operator to the collision danger and would provide him with safe means to avoid a collision.
- Multiple target avoidance e multiple target avoidance testing procedure would evaluate the algorithm under the AUTO/FLYTO flight mode with the AP04 auto pilot assuming full control over the UAV flight operations. The main objective is to provide a safe waypoint rather than actual flight instructions as per the single target avoidance.

CHAPTER 8

TEST PERFORMANCE RESULTS

8.1 Single target avoidance

The single target avoidance testing procedure comprises of providing the trajectory angles of the aircraft and Snark to the algorithm. To provide the simulation program with the aircraft trajectories, the program upon start up would request the speed and heading of each aircraft and the aircraft location relative to the Snark as described by range and elevation and azimuth angles.

The values provided to the program are as follows:

- Range to the target aircraft
- Azimuth detection angle (α)
- Elevation detection angle (β)
- Velocity of the target aircraft
- Azimuth heading angle of the aircraft (γ) relevant to the heading of the Snark (with 0° being forward)
- Elevation heading angle (δ) of the aircraft: This determines the climb or the descent of the aircraft.

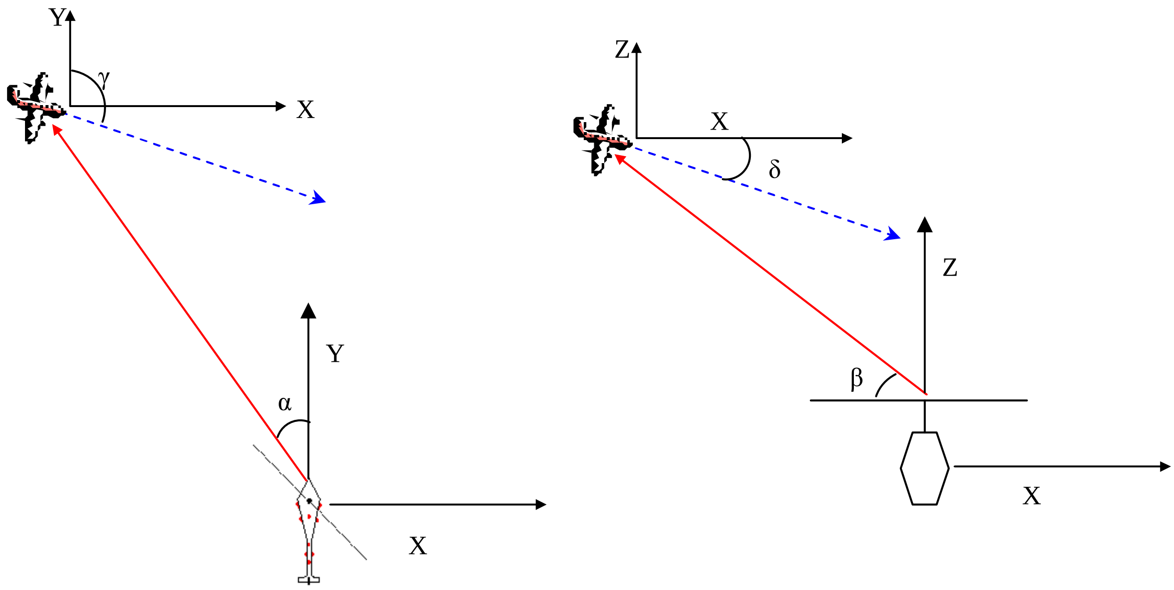

Figure 18 - Representation of elevation and azimuth angles

The results of the testing process are divided into three sections, each corresponding to a typical type of avoidance manoeuvre. This simplifies interpreting the results, and provides greater insight into the validity of each avoidance manoeuvre discussed in earlier chapters (Namely: changing speed, changing altitude, and changing the heading).

It is perhaps worthwhile to note that fewer scenarios than expected caused the two aircraft to be less than 400m apart. This is mainly due to the noticeable difference in cruising speeds of the Snark as compared to fast moving jet aircraft. As a matter of fact, in most cases the flight path of the aircraft will harmlessly pass across the Snark's travel path with separations larger than 400m.

8.1.1 Results of Changing Speed Manoeuvres

As predicted during the algorithm design, avoiding a collision via changing the velocity of the Snark provides the most efficient and simplest means to avoiding a collision. This is confirmed as will be demonstrated below.

To demonstrate the response of the algorithm, the following collision scenario will be discussed:

The radar detects an aircraft at range of 10km, with an azimuth angle of -50° (to the left of the Snark heading) and at the same flight level as the Snark (i.e. zero degrees of elevation angle), with the Snark travelling at 100kts and the aircraft at 500kts

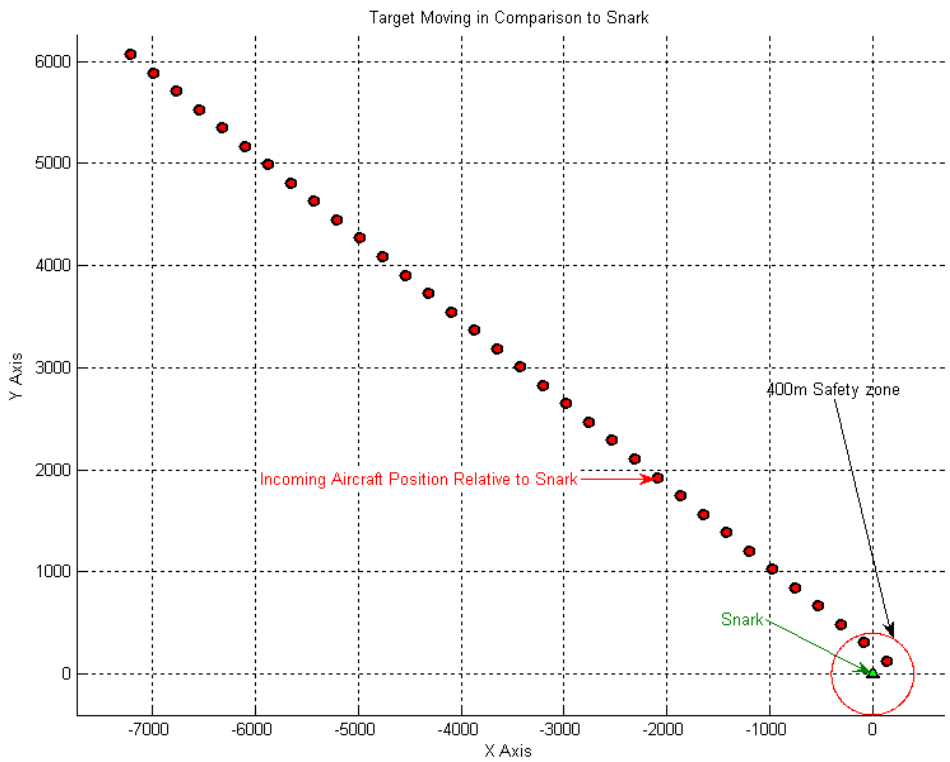

Figure 19 - demonstration of target approaching the Snark

In this scenario, the algorithm detects that the range is decreasing with time, thus it initialises a request to estimate if the target (at the current trajectory of both aircraft) will be less than 400m away from the Snark. The results of the estimation are as follows:

- Closest distance of approach (CDA) is 183.55m
- Estimated time to closest approach is 34 seconds
- The algorithm also suggests either to increase the speed to approximately 146kts or to reduce the speed to approximately 84kts (As per Equation 11). The calculation estimates that adopting any of the two values would increase the separation distance to 400m as required.
- However, since the maximum speed of the Snark is 145kts, the algorithm would instruct the flight controller to reduce the speed to the lower value of 84kts
- Having adopted the new velocity, the Snark would require approximately 4 seconds to reach the safe velocity and an extra second to stabilize its path.
- Please refer to Figures 20 and 21 for a visual representation of the manoeuvre, also please note that the units of X and Y axes are in meters.
-

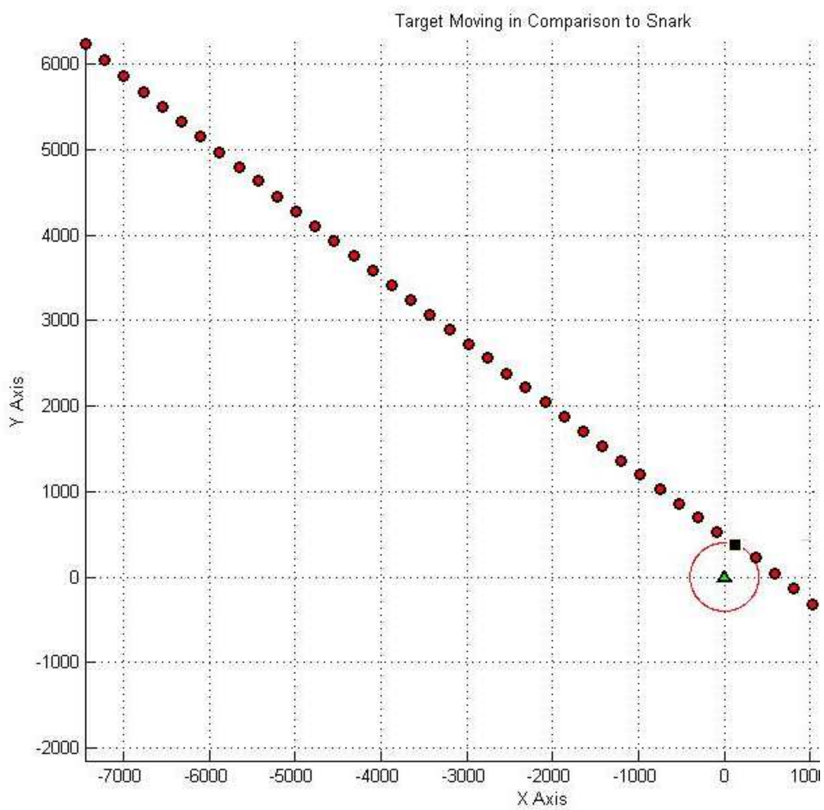

Figure 20 - Target trajectory after changing Snark velocity

As can be seen from the figure above, the target/aircraft just passes at the 400m safe limit. As can be seen from the figure, this method provides the least change to the flight path of the Snark, and can be extremely effective in congested air traffic, as there will be no need to change course and endanger other aircraft in the process.

The following table represents a sample of other cases of this algorithm:

α	β	γ	δ	CDA	TCA	Tman	Final CDA	Action
-50	0	120	0	183.55m	32 sec	5sec	400m	Reduce V
160	0	-15	0	231.94m	48 sec	6sec	400m	Reduce V
200	0.4	15	0	242m	47sec	3sec	400m	Reduce V
-80	0.5	90	0	244m	38sec	3sec	400m	Increase V
165	0.45	-12	0	70.17m	47sec	9sec	400m	Reduce V
175	-0.45	-5	0	231.4m	48sec	11sec	400m	Increase V
160	-0.4	-15	0	242.21	48sec	7sec	400m	Reduce V
190	0.5	-10	0	440m	N/A	N/A	440m	N/A
170	0.5	-8	0	98m	48sec	10sec	400m	Increase V
-15	0.45	160	0	307m	32sec	3sec	400m	Reduce V

Table 2 - Test results of velocity change

During these tests, several elevation values were introduced to the algorithm to measure their effects on the working of the algorithm. It was discovered that, elevation detection angles (α) of more than 2° would result in a no collision scenario (as the altitude separation would be approximately 400m), and increasing the angle beyond 0.7° would result in a very slight change to the Snark's velocity during this evasion manoeuvre.

The cases shown in Table 2 are not the most severe collision scenarios; however, they are quite varied and give an impression of the algorithm's performance according to a wide range of approach trajectories.

In some of the scenarios above, the value N/A represents a no-avoidance action required, as the separation between the incoming aircraft and the Snark is larger than the 400m safety limit.

8.1.2 Results of Altitude Change

Changing the flight level or altitude of the Snark to avoid a collision is the second method to be considered for the avoidance algorithm. Changing flight levels is the prime method used for civilian aircraft to avoid collision under the TCAS system.

However, the slow rate of climb of the Snark (5m/sec for altitudes under 10,000ft and 3m/sec for altitudes above that) would limit using this method.

To demonstrate this, the following simple test values show how limited this action is:

Assume that the radar system detects an aircraft travelling at 500kts at a range of 10km, and it is approaching the Snark from a head on angle (α = 0° with γ = 180°). However, the aircraft path is slightly above the Snark, and the radar system registers an elevation angle of 2.3° (β = 2.3°).

Again, the algorithm detects that the aircraft is approaching the Snark and analyses its path. A collision scenario is detected, and the aircraft will approach the Snark to within approximately 366m at time = 32sec.

The avoidance algorithm calculates and estimates that changing the velocity will not resolve the situation since the trajectory of the aircraft is directly opposite to that of the Snark and changing the velocity will only change the time at which the collision would occur rather than the separation between the two trajectories. The output of Equation 11 in this case would be two negative values, instead of low or high positive velocity values.

The algorithm would evaluate changing the altitude as a solution, by using Equation 12 to calculate two altitude values: a higher and lower altitude value.

For the particular collision scenario above, the algorithm will calculate two safe flight levels, one 33m below the altitude of the Snark and one 766m above.

The program will then calculate how long the Snark will take to reach these altitudes depending on the climb/descent (in relation to the current altitude).

In this case it would take approximately 7 seconds to reach the lower altitude. Therefore the aircraft would descend to avoid collision.

This however, exposes the limitations of changing altitude of the Snark to avoid a collision, where the time to closest approach (TCA) and the climb/descent rates are the determining factors.

The following is an expansion on the above scenario: where we kept the same headings, velocity and trajectories, but have changed the elevation angle, and in turn the safe altitude values.

β	CDA	TCA	Altitude Low	Altitude High	Action	Time to Alt
2.1	366.44m	32	-33.58	766.44m	Descent	6sec
2	346.99m	32	-51.02	748.98	Descent	10sec
1.8	314.11m	32	-85.89	714..11	Descent	18sec
1.5	261.77m	32	-138.23	661.77m	Descent	28sec
1.2	209.5m	32	-190.58	609.42	Descent	36sec

Table 3 - Test results of altitude change

As can be seen, changing the altitude is limited to aircraft approaching (assuming they travel in a level trajectory) at an initial elevation angle of detection greater than 1.2°. The minimum vertical separation that this algorithm can evade is approximately 215m, else the Snark would not have adequate time to reach the new altitude.

Other scenarios have been tested, such as aircraft changing altitude and their descending or ascending trajectories intersecting with the Snark's trajectory, however changing the velocity of the Snark proved to be the more efficient of the two methods, and in extreme cases turning or changing the trajectory of the Snark is the most effective means, as described in the next section.

8.1.3 Results of Turning

Changing the heading of the flight path (turning the Snark) is as an effective means to avoiding a collision whenever varying the velocity of the Snark would not avert the collision. However, it is by the far the most demanding of the three methods described. It also involves the most complex manoeuvre structure in terms of direction control and range estimation, and it is the most dependent of the three methods on the UAV's manoeuvring capabilities and performance.

Factors such as the rate of turn, banking angle of the aircraft, and the turning radius must be considered for an accurate manoeuvre performance.

Some of these factors can be calculated and set mid flight to vary the turning radius and turning time of the process. From Equation 13 and 14, we can see that both the ROT and the turning radius are dependent on the value of the banking angle. Therefore the accuracy of the turn, i.e. the length of the turning curve, and consequently the location of the Midway point and the final horizontal offset distance, rely directly on the turning radius and the banking angle.

Under normal circumstances in a manned aircraft, the value of the banking angle can be varied, to change the value of the ROT, Varying the banking of an aircraft can be selected by the pilot to achieve an optimum ROT and turning radius for a more accurate flight trajectory, however this is not the case of a fully-autonomous UAV.

In this experiment, we have decided to use a single value for the banking angle through out the duration of the turn. During initial testing of the algorithm, it was clear that a banking angle of 25° presented an optimum solution in terms of turning (and final separation accuracy), response time, and load factor (or G-Load) on the airframe [30]& [40]. This argument is demonstrated in Table 4, as choosing a banking angle of 25° results in excellent average avoidance times, more accurate separation values on average, and tighter turn radius.

However, that increasing the banking angle beyond 27° would increase the load factor on the airframe, and would require the engine to increase its output power accordingly to maintain flight altitude and turning speed. [40] & [46]

These scenarios were conducted with a Snark speed of 100kts, aircraft approaching at 500kts, and travelling at an angle of 180° opposite the Snark.

α	Bank = 20deg		Bank = 22deg		Bank =25deg	
	CDA (m)	Time Sec	CDA (m)	Time sec	CDA (m)	T sec
0	400.6	20	412.23	28	400.58	16
0.25	402.4	22	403.74	22	411.4	24
0.5	401.22	27	418.68	31	409.49	21
0.75	411.14	24	400.07	25	406.76	27
1	400.19	29	408.28	15	405.6	18
1.25	400.31	18	403.64	17	401.52	19
1.5	406.16	19	401.86	15	408.81	12
1.75	404.41	14	403.2	12	404.13	10
2	402.17	8	400.02	7	412.56	12

Table 4 - Effects of banking angle on turning performance

Further test cases also showed that turning/changing the heading is more effective in some severe cases where an aircraft approaches the Snark at almost head on values, usually within ±10° of the y-axis of the Snark trajectory. In some of these cases, changing altitude is not suitable and the change of speed needed to over take or allow the aircraft to pass exceeds the Snark's performance.

Expanding the test scenarios, we get the following values

α	β	γ	δ	CDA	T man
0	0	180	0	400.58m	16sec
1	0	180	0	405.6m	18sec
-1	0	180	0	405.6m	18sec
179	0	0	0	405.6m	18sec
181	0	0	0	405.6m	18sec
175	0	-5	0	596.30m	20sec
0.5	0	180	0	409.5m	20sec

Table 5 - Turning results at various approach angles

The time value (T man) is an indication of the time that the Snark takes to identify the threat, calculate the manoeuvre, perform the initial turn to reach the midway point, perform the second turn of the avoidance manoeuvre, and stabilize the path of the aircraft after ensuring that the final separation is in excess of 400m.

The main concern when using this type of avoidance manoeuvre is to increase the gap between the two aircraft to more than 400m as quickly as possible as compared to the previous two methods.

This is mostly due to the fact that the ROT and the turning radius are rather constant due to the preset value of the banking angle.

At 25° the ROT is 0.09Rad/sec and this translates to a rate of heading change of 5.09 degrees/sec. This sometime would result in the Snark performing extra miniature turning manoeuvres during the stabilization part to ensure a separation of at least 400m. We can see that in the case of the target approaching from an angle of 175° as per Table 5 above, as the Snark would turn to the right to avoid the target, the algorithm would continue evaluating the two trajectories and it would maintain the course of the Snark at a distance of 596m rather than 400m as reducing this distance would turn the Snark back into the aircraft trajectory which is not recommended as the time remaining is perhaps not sufficient for a second complete turning manoeuvre. The algorithm would consider that the separation albeit larger than required is still considered safe and would not compromise changing it. The main cause of such an excessive separation distance is that the combination of the target speed, heading relative to Snark, Snark travel path and speed form a unique case where the target remains on a collision course even after the first turn (separation distance is 390m after the first avoidance turn), and safe separation is not assured until the Snark moves performs a second turning manoeuvre to avoid it. Please bear in mind, that the turning manoeuvre is a last resort, and a combination of a turning and velocity change avoidance manoeuvres would ensure a better separation value.

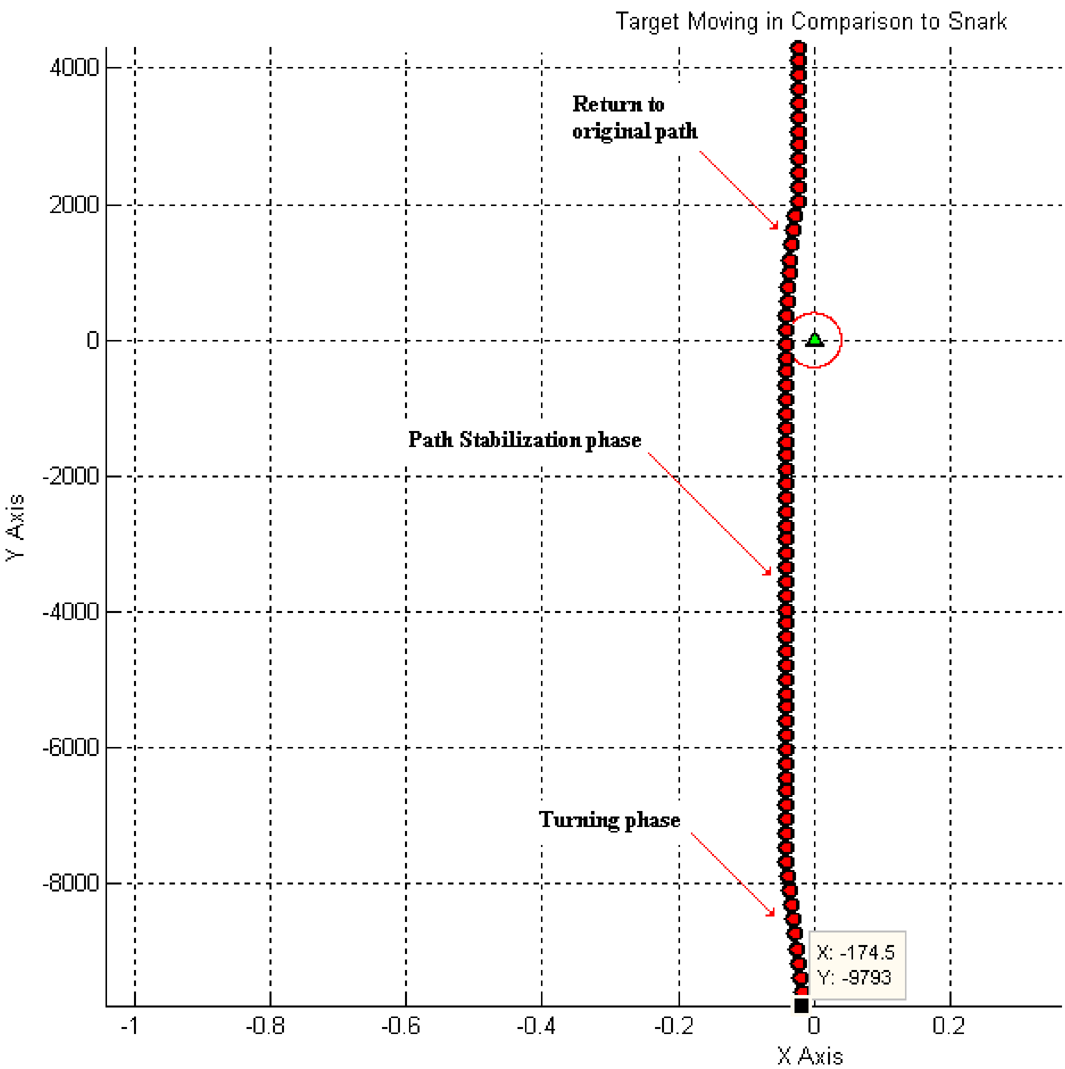

Figure 21 - Representation of a Snark turning manoeuvre

8.2 Multiple Aircraft Avoidance

To avoid multiple aircraft, the main emphasis of the algorithm shifts from maintaining a safe path while performing the simplest manoeuvres, to finding a safe waypoint and flying to it within the time constraints defined by the least estimated time to collision.

As explained in section 5.8, the algorithm estimates the relative position of the intersection of the two aircraft and then the best of the four possible safe waypoints is estimated. However, the separation distances at the safe waypoint may (and usually would) exceed the 400m limit.

This extra safety margin is necessary to ensure that the algorithm finds a safe waypoint as fast as possible, and more importantly it reduces the complexity of the algorithms and programme code and ensuring a reliable and a safe solution is found.

Clearly, the number of aircraft that appear on the radar screen must be limited to a sensible and a realistic number and assuming that all other aircraft are also obeying standard flight rules.

During the tests conducted by NASA during the ERAST test stages, Wolfe noticed that attempting to avoid more than 3 aircraft at a time would prove to be an extremely demanding task for an autonomous aircraft operating without the benefit of ATC support. [44]

Therefore, we shall conduct the multiple aircraft avoidance test with only 3 aircraft sharing the same air space as the Snark.

Also, we will assume that the velocities of these aircraft are different to each other, and would have a value between 400kts and 500kts, again simulating actual aircraft travelling at typical speeds.

Although the algorithm is based on the 3D mathematical equations and includes the altitude and climb rates of all aircraft involved, it has been restricted to finding waypoints in the horizontal (XY) plane.

This is due to the fact that the Snark's climb rate is very low, as demonstrated in previous sections and test results. It is preferable to fly to a waypoint on the same plane as it is always faster than climbing to a waypoint on a different altitude. This would also reduce the possibility of infringing on different flight levels.

One must consider test scenarios that allow for a third aircraft that enters the radars sensing range while the Snark is performing an avoidance manoeuvre. This can be quite dangerous if the Snark's current avoidance path (or the waypoint) lies in the path of the third aircraft.

To demonstrate the working of the algorithm, we considered the following scenario:

The Snark initially detects two incoming aircraft both at approximately 10kms distance and both aircraft have the same altitude (flight level) as the Snark, with no climb or descent velocity components, the paths of the two aircraft were as follows:

Aircraft	Azimuth (α)	Heading (γ)	Velocity (kts)	Range (km)
1	0°	180°	500	10
2	-50°	120	400	10

Both aircraft are on collision paths, and the t_Q (time to collision) is approximately 31 seconds for aircraft 1, and 41 seconds for aircraft 2.

The collision avoidance algorithm evaluates the path of both aircraft, and it determines that a safe waypoint exists within the time constraints, and that the safe waypoint location is at x = -400m and y = -389.15m and it is located at a distance of 400.15m away from the current Snark.

The coordinate values of the waypoint (x, y, and z) are all relative to the intersection point as explained earlier. The algorithm calculates the total time to reach that waypoint as 7.78seconds and as we can see in the figure below, that waypoint lies in the lower quadrant of the intersection where it is the closest safe waypoint to the Snark’s estimated location if no avoidance measures were taken.

Next, the algorithm calculates the heading and velocity required to reach that waypoint from the current position. Once the heading calculation is performed, the information is passed to the flight controller in FLYTO mode. The Snark/flight controller would change its heading and speed to reach that specific waypoint. While heading to the designated waypoint, the flight controller checks if one of the aircraft changes its direction, or if the Snark has reached that waypoint. In Figure 22, the blue dotted line indicates the flight path of aircraft 1 while the red dotted line indicates the flight path of aircraft 2.

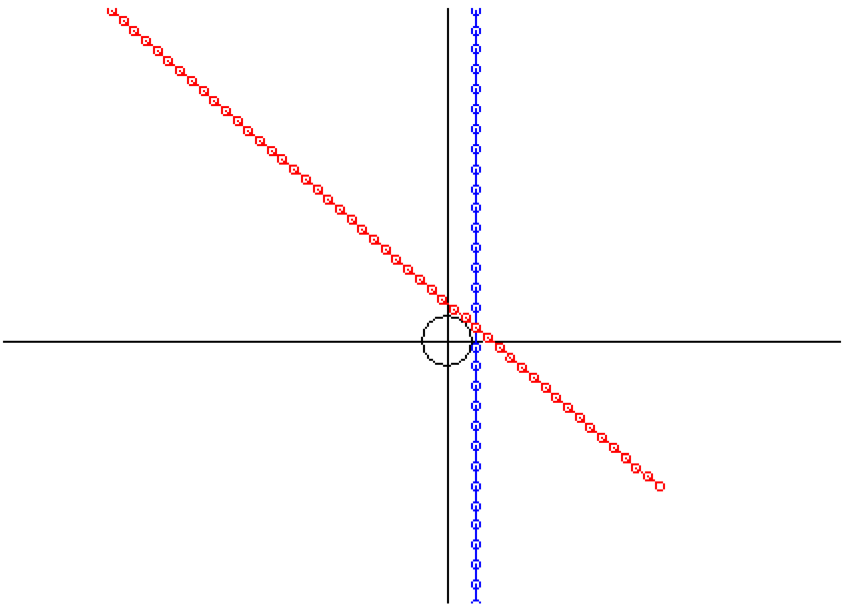

Figure 22 - Avoiding two aircraft

To add a slight complication to the working of the scenario, we introduced a third aircraft (indicated by the green dotted line in Figure 23) that appears on the radar scan, as the Snark is changing its heading to reach that waypoint.

The flight path of the third aircraft is as follows:

Aircraft	Azimuth (α)	Heading (γ)	Velocity (kts)	Range (km)
3	1°	180°	450	10

This new aircraft has an approach CPA of 219m and a TCA of 34 seconds as compared to the current Snark speed.

As we can see, the path of the third aircraft would render the safe waypoint useless, as it lies directly in the path of aircraft 3. This new aircraft would cause the program functions responsible for checking the safety of the current path to indicate that the path is no longer valid.

Hence the algorithm would repeat the evaluation process to seek a new safe waypoint or a safer path, this time including the path parameter, range and TCA time of aircraft 3.

The paths of aircraft 1 and 3 are parallel, so the algorithm cannot calculate any intersection point however it can find a safe solution (another waypoint) by repeating the process as above to find a more suitable waypoint. The algorithm calculates four possible waypoints for each pair of aircraft, i.e. 1 & 2, 1 & 3, and 2 & 3 and then the algorithm selects the most suitable one. If no safe solution exists using that method, then it uses either the changing speed or turning avoiding processes to find a new waypoint location and a new speed.

In this case, the algorithm estimates that a new waypoint is required. In this case the new waypoint requires the Snark to adjust its path and adopt a heading of 81° to avoid a collision with this new aircraft.

If the paths of two aircraft are parallel and do not intersect (for example aircraft 1 and aircraft 3), the algorithm uses the turning avoidance process calculation to find a way point to the left or to the right of these two pairs of aircraft. In this example the waypoint is one of the four points calculated around the intersection point of aircraft 1 and 2 (please refer to Figure 23). This new waypoint was approximately 800m away from the intersection point of aircraft 1 and 2, and it provides the best solution as it is the closest waypoint to the current location of the Snark.

The Snark required approximately 15 seconds to predict a collision, calculate a new waypoint, and to reach this new waypoint.

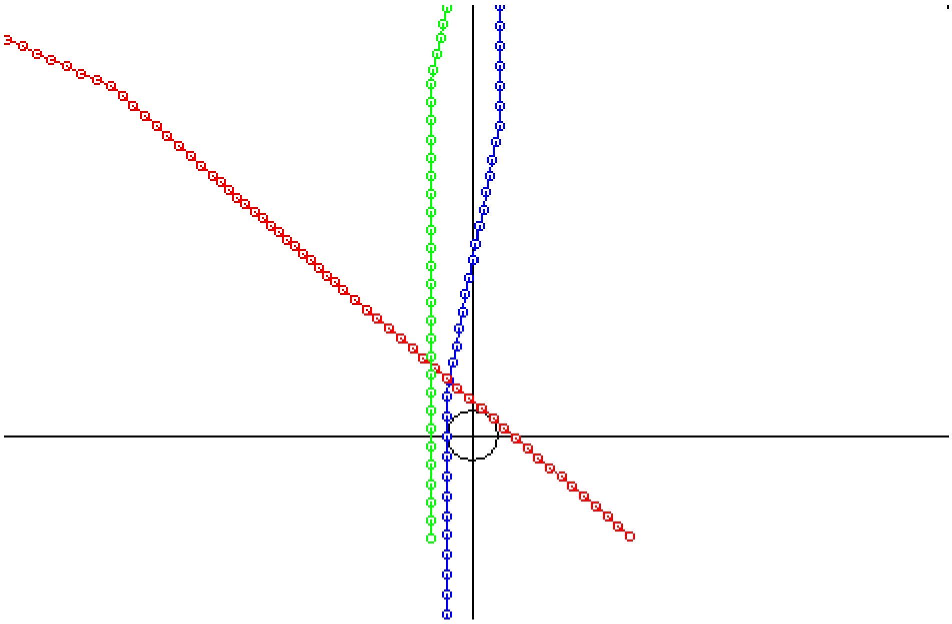

Figure 23 - Changing waypoint to avoid a third aircraft

It can be seen that this algorithm is ineffective against parallel targets. Therefore it is required that this algorithm is used in conjunction with the two previous avoidance processes (namely changing speed and turning) to ensure a reliable operation.

In some cases, the algorithm will find a suitable waypoint, but it requires an abrupt change of direction in excess of 90°, i.e. the new waypoint requires the Snark to turn almost immediately in mid air. Although this might be achievable under certain circumstances, it would subject the aircraft to high levels of G-loading (or structural stresses) which is not preferred. Therefore the waypoints are also evaluated according to the heading they require. For example if a waypoint requires an excessive change of heading it would not be used and a more suitable waypoint would be used, even if the other waypoint is further away.

Table 6 contains some further test cases of this method.

Air 1	Air 2	Air 3	dir WP1 (deg)	dwp1 (m)	dir WP2	dwp2 (m)
(0, 180)	(-50, 120)	(-3,180)	-88.4	400.15	-45	1020.8
(0, 180)	(-50, 120)	(3, 180)	-88.4	400.15	-45	529.4
(1.5, 180)	(-50, 120)	(-1, 180)	41.43	1000.16		
(-80, 80)	(50, -120)	(-4, 180)	0	2511.9	-45	1083.9
(1.5, 180)	(-50, 120)	(-1.5, 180)	41.43	1000.16	-45	1178.6
(1.5, 180)	(0, 120)	(-1.5, 180)	-45	195.5	-45	878.2
(1.5, 180)	(0, 120)	(-178, 0)	-45	195.5	-45	1050.8
(1.5, 90)	(0, 120)	(-178, 0)	-45	565.7		

Table 6 - Multiple aircraft test results

In examining the third test case in Table 1Table 6, the scenario indicates two aircraft approaching the Snark at comparable approaches to the firs case, except with aircraft 1 (blue dotted line) flying in a path approximately 262m to the left of Snark heading.

The algorithm would recognize the collision scenario and would calculate a safe way point that would satisfy both aircraft at the following location: x = 661.8m and y = 749.9m and is approximately 1000.16m away from its current location, and the waypoint lies at a heading of +41.43º from the 0º heading of the Snark. The Snark also estimates that it has 31 seconds to reach that waypoint,

Whilst the Snark is heading for that waypoint, we introduce aircraft 3 (represented by the green dotted line) to the scenario. We decided that aircraft 3 is flying at a path of -1º of the side, and in opposite direction to the flight path of the Snark.

The introduction of the new incoming aircraft would prompt the algorithm to recalculate whether the current path is safe. And in this case the current path of the Snark (heading towards the initial waypoint of (661.8m, 749.9m) can also provide a safe solution for the new incoming aircraft with no need to adjust the value of the waypoint, as can be seen in Figure 24. Thus the algorithm does not provide any secondary waypoint into Table 6.

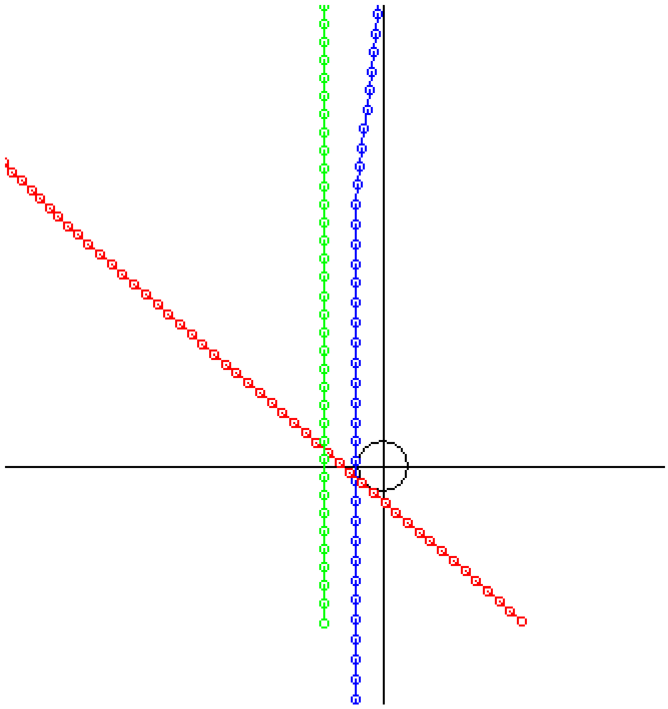

Figure 24 - A safe path is provided by a single waypoint

CHAPTER Error! Reference source not found.

DISCUSSION

The test cases described in the previous chapter give an indication of how the program performs under diverse circumstances.

All of the considered collision scenarios (especially under the single target avoidance) have illuminated a known fact, that changing altitude as an avoidance measure should only be instigated when the vertical separation is greater than approximately 280m.

As can be seen from Figure 25 below, the limited climb and descent rate of the Snark limits using this type of avoidance manoeuvre to targets passing through the shaded zones as presented, although the actual volume of that limit zone would depend on the trajectory angles of the aircraft (if it is in climb or descent or at an offset horizontal angle).

The values presented in the figure are quoted assuming that the Snark is travelling at an altitude above 10,000ft (limiting its climb or descent rate to 3m/sec), a target detection range of 10km with the target travelling at a speed of 500kts.

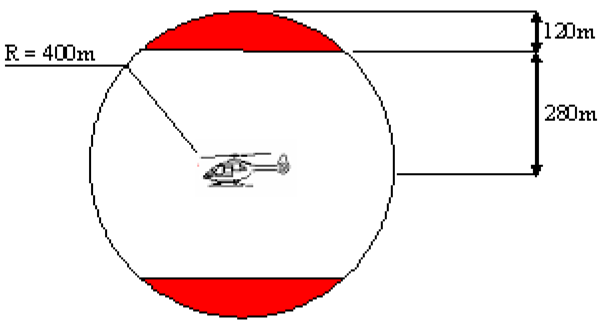

Figure 25 - Safe zones during altitude change manoeuvres

From this, it is safe to assume that changing altitude to avoid a collision is perhaps the least used manoeuvre throughout this experiment, as it cannot be relied on to avoid the more severe cases.

During the single target avoidance, it was quite clear that the two main methods to be used are: changing velocity and changing heading. The most efficient method of the two is varying the velocity of the Snark and increasing the separation gap between the two aircraft. The results show that it is the faster of the two to deploy, and one can achieve safe distance separation in less than 10 seconds.

As mentioned earlier, turning/changing heading provides the Snark with a guaranteed collision resolution, and a minimum separation distance of at least 400m.

Both of these methods are flexible and can provide a safe flight path even with targets that change their approach velocities. In the case of a turning algorithm, a target has to change its direction intentionally towards the Snark to cause another collision alarm.

From the test results, we can more or less determine that TAs caused by targets approaching from head on or near head on angles (approximately ±15°) of the y-axis of the Snark are avoided by turning manoeuvres, and targets approached from angles exceeding ±40° can be resolved via changing velocity.

Some specific cases were noted that can be cross-resolved so to speak by the two algorithms. For example, the case of a target approaching from 175° and with a travel path of -5° (i.e. the target is approaching the Snark from aft, and its travel path has a slight offset to the left). In this case, both changing the velocity or turning methods can and will resolve the situation. However, in some cases, the algorithm requires the Snark to accelerate beyond its maximum dash speed, and therefore changing direction is required to guide the Snark to a safety point.

It is worthwhile to mention, that the target's speed of approach or the range of detection has a greater effect on the evasion process for a change of speed manoeuvre than for a turning evasion manoeuvre. Whereby the difference between upper and lower calculated Snark safe speeds would increase as the Snark's distance to the point of closest approach decreases.

This is evident in multiple target avoidance, where it was noticed that increasing the target velocity to beyond 700kts (approximately 1300km/h) would radically limit the response of the Snark, and would reduce the number of possible waypoints.

Initially, the program estimates the path of all detected aircraft and then the algorithm calculates the four safe way points that the Snark can us to avoid a collision. However, we discovered that one or more of these waypoints (depending on the approach angles and speeds of the aircraft involved) will be too far away for the Snark to reach at its current speed. The algorithm then calculates the velocity needed to reach that specific waypoint within a time limit equal to the least TCA of the two approaching aircraft, i.e. reach that far waypoint before the first aircraft reaches minimum distance.

During the testing phase, we noticed that calculating a new Snark velocity would ultimately change the geometry and the distances to that specific waypoint (without affecting the equivalent values of the other waypoints). This effect can be attributed to the fact that changing the velocity of the Snark would affect the value of the relative speeds of the aircraft used in the time and distance estimation equations as explained in section 6.5.

Whereas if we allow the algorithm to calculate a travel velocity for the Snark for each waypoint, by estimating the velocity needed to traverse across the distance to waypoint and dividing it by the time required to reach that waypoint, this can result in remarkable reductions in the times and travel distance to reach previously waypoints that were too far away. One example in the case where there are two aircraft, with one approaching the Snark from an angle of 0° (travelling at 180° at a speed of 500kts) and the other approaching the Snark at an angle of -50° (travelling at an angle of 120° at a speed of 400kts). Initially the algorithm identifies 4 possible safe points, but the algorithm excludes one of them as it is at a distance 3300m away from the Snark and it would take 66 seconds to reach, such that the Snark would never reach that waypoint in time and the time to reach that waypoint is calculated and it was the larger than the least TCA time of 31seconds.

However, if we increase the velocity of the Snark and recalculate the relative velocity values used throughout the algorithm, the time taken to reach that point is reduced from 66 seconds to 31 seconds while only increasing the velocity from 100kts to 102.4kts!

It is important to note that certain cases may appear dangerous and without a solution while the Snark is performing a manoeuvre. To fully explain this, please consider the third test case of the multiple avoidance tests (refer to Table 6). In this case the Snark is required to avoid two incoming aircraft (aircraft 1 approaching the Snark from a 1.5° heading angle and it is travelling at 180°, and aircraft 2 which approaches the Snark from an angle of -50° travelling at a heading angle of 120°).

The program calculates the locations of the waypoints, and selects a waypoint that is not on the path of any incoming aircraft. However, due to the fact that the appearance of a third aircraft 16 seconds later (while the Snark was en route to the initial safe waypoint to avoid aircraft 1 and 2) had caused the program to abandon this manoeuvre because the avoidance flight path the Snark was undertaking is dangerous to pursue.

The astonishing fact about this case, was that the third aircraft passes to the right of the Snark in a straight path, and the distance from that path to the original Snark flight path (0° line) was 1464m Furthermore, the Snark was required to change its location by approximately 660m to the right of the original path line, as that was the x-coordinate of the waypoint. This still left more than 400m of separation between the Snark and aircraft 3. However the algorithm detected a collision scenario between the two!

This was attributed to the fact that the algorithm estimated that if the Snark maintained its current course (while performing a manoeuvre), it would be in a dangerous situation as aircraft 3 would be heading towards the Snark (refer to Figure 26) even though it did not presented any real danger.

This was remedied by modifying the safety check function to the algorithm when the Snark is evaluating the suitability of each waypoint. This modified function would estimate the separation between the Snark and all other aircraft while the Snark is heading to the waypoint (by using the avoidance velocity and heading), and then it estimates the separation between the Snark and other aircraft after it reaches the waypoint (by using the original Snark velocity and heading but adjusting the location of the Snark to that of the waypoint).

Having done that, the third aircraft did not present any danger once the course was adjusted.

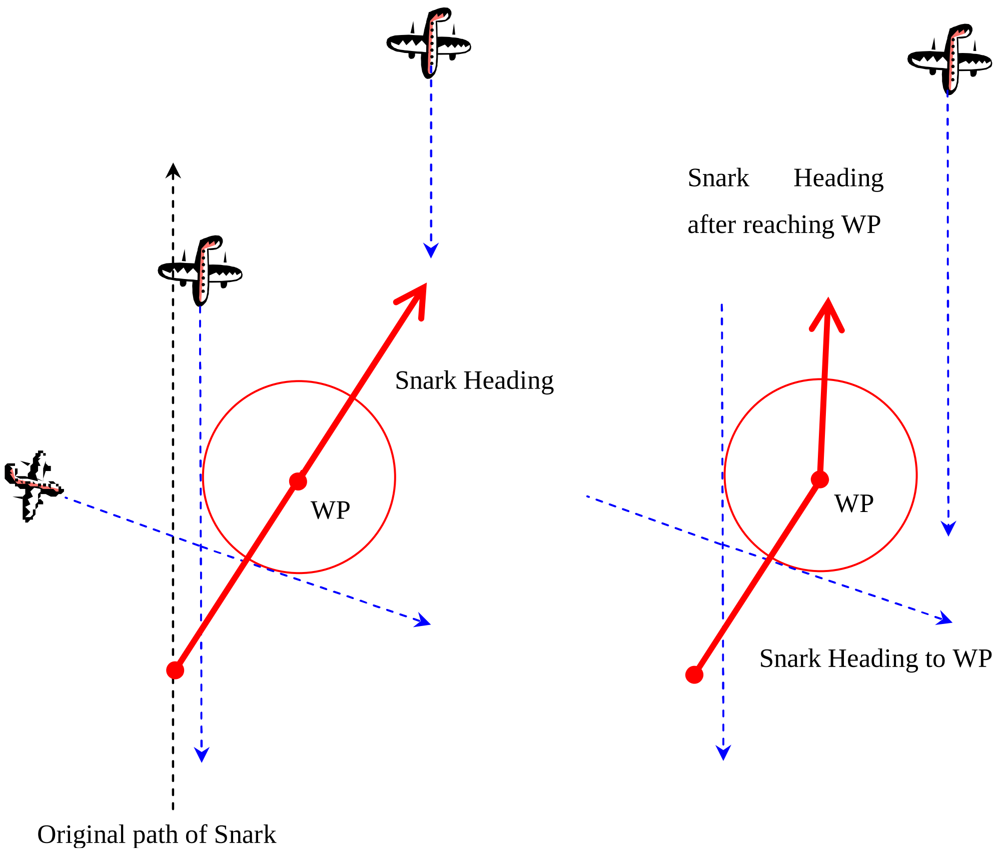

Figure 26- Path estimation after reaching WP

It is perhaps important to note that whilst it is a pre-requisite to adhere to aviation standard avoidance manoeuvres (the CFR 91.113 flight rules); however, these rules may become impractical when avoiding multiple aircraft (or multiple obstacles for that matter). The differences in the applications of UAVs and manned commercial aircraft would necessitate defining new rules for low flying UAVs.

CHAPTER 9

CONCLUSION AND FUTURE WORK

The main objective of this research was to develop an automated collision avoidance system that can be installed on an unmanned aircraft to enable it to fly un-assisted into non-regulated airspace. The work included the simulation of the key parts of this proposed system, namely the radar, the flight controller, and the algorithm/collision avoidance unit.

The overall performance of the system has been satisfactory, and the test results obtained for the different avoidance actions have demonstrated that the software design concepts and the avoidance schemes are correct. The system was tested under different real and exaggerated intrusion conditions, and under different aircraft travel speeds, at which the algorithm performed with acceptable results and was able to find a solution for all test cases.

From the results obtained throughout this experiment, we conclude that changing the velocity and/or travel direction of the Snark represent the most efficient and safest avoidance actions respectively to resolve single aircraft collision scenarios.

Also, to resolve collision scenarios with up to three aircraft, we showed that flying to specific safe waypoints is the most suitable option, where the objective of the algorithm would be to estimate a safe set of coordinates or a travel path to stay clear of a limited number of intrusive aircraft.

Although the results of this experiment have been satisfactory thus far, it would prudent to state that tests should and must be conducted under real life conditions. Furthermore, in order for this system to gain the proper approval and the design to be delegated for phase III testing (select limited commercial in supervised environments), the system must be modified to take into account real-time applications, signal delays, data transfer errors and delays, processing limitations of hardware modules, and also to allow the system to be flexible enough to interface to different flight controller and radar designs.

For future work, we propose that an avionics test suite to be purchased, this should include a flight controller unit, an SBC host for the avoidance system, and a radar system simulator. Almost all avionics systems manufacturers provide such simulators and training solutions, although at high costs.

However this cost can be avoided if a joint venture is formed with willing avionics manufacturers and such algorithms eventually be installed on their system as an added option.

Perhaps most importantly, we must develop more complicated simulation models for target aircraft, such that we can take into account aircraft targets that are flying in a non-straight path, (turning aircraft) and aircraft that can vary their travel velocities mid-flight. The purpose of this is to monitor the effects of such occurrences and to evaluate the resulting calculation errors on performance of the algorithm.

Applications using UAVs may require them to fly between multiple obstacles or between buildings and may even see them travel in swarms for reconnaissance or search and rescue purposes, all of these applications would require the collision avoidance algorithms to follow more flexible avoidance rules rather than the rigid CFR 91.113 rules, that were originally designed for resolving collision paths for civilian aircraft flying on defined paths..

APPENDICES

This thesis includes a CD that contains all the source code for the collision avoidance program (both the initial Matlab simulation model and the final simulation C-code), in addition to schematic diagrams and some selected results of test cases. To run these programs you will need a computer system equipped with a Microsoft 2000 or XP, and running on an 80x86, Pentium class, or AMD class processors.

To run the executables, you will need to have a Matlab environment for the Matlab files naturally, and a Borland C++ environment for the simulation C code.

To run the C files, either create a new project under Borland IDE or use the existing Borland Project file (MultiAvoid.bpr), then compile, build, and finally run the project.

Once the executable file is running, you will see a radar plot with the UAV in the centre, and an entry table on the right. There you can enter the velocity, direction, and azimuth and horizontal angles of the approaching aircraft.

It is recommended that one tries some of the test cases listed through out this document to acquire initial expertise with the code then the examiners can devise other cases as they wish.

REFERENCES

[1] Eby MS, Kelly WE. Free Flight Separation Assurance Using Distributed Algorithms. Aerospace conference; 1999 6 - 13 March; Aspen, CO; 1999. p. 429 - 41.

[2] US Department of Transportation. A ROADMAP FOR DEPLOYING UNMANNED AERIAL VEHICLES (UAVs) IN TRANSPORTATION. Santa Barbara, CA: NCRST, FAA, and University of California; 2003 2nd December.

[3] Colligan W. Friendlier Skies. The Washington Post. 2004 12th February.

[4] Law J. RAs and 1000ft level-off manoeuvres. ACAS II bulletin. 2003 March(2):1 - 3.

[5] Ballin MG, Hoekstra JM, Wing DJ, Lohr GW. NASA LANGLEY AND NLR RESEARCH OF DISTRIBUTED AIR/GROUND TRAFFIC MANAGEMENT. Los Angeles, CA: American Institute of Aeronautics and Astronautics; 2002 1 - 3 October.

[6] FAA. Reduced Vertical Separation Minimum. [Government reference website] 2006 [cited 2007 25th March]; Available from: http://www.faa.gov/about/office_org/headquarters_offices/ato/service_units/enroute/rvsm/

[7] Hayes PB, Mahon T. Europe's fresh focus on human factors. Aersopace America. 2005 October:4 - 8.

[8] DeGarmo MT. Issues Concerning Integration of Unmanned Aerial Vehicles in Civil Airspace. Research Report into UAV technology under Contract from FAA. McLean, Virginia: MITRE, Centre for Advanced Aviation System Development; 2004 November.

[9] Jardin MR. Real-Time Conflict-Free Trajectory Optimization. *5th USA/Europe ATM R&D Seminar*. Budapest, Hungary: NASA Ames Research Centre 2003.

[10] Ryan A. An Optimized Trajectory for a UAV to Track a Helicopter. Progress report. California: UC Berkeley; 2004 11 December.

[11] Weatherington DD. Unmanned Aircraft Systems - Road map. Washington D.C. : US Department of Defence; 2005.

[12] Bernier R, Bissonnette M, Poitevin P. MMW Radar for Non-Cooperative Collision Avoidance. *UVS Tech 2004*. Brussells, Belgium: Amphitech Systems International 2004.

[13] Lee HC. Implementation of Collision Avoidance System Using TCAS II to UAVs. *24th Digital Avionics Systems Conference*: IEEE 2005.

[14] Federal Aviation Authority. Introduction to TCAS II Version 7. In: Transportation Do, ed. Washington DC: US Department of Transportation 2000:1 - 45.

[15] Mid-air collision. Flight Safety Australia. 2004 2004:22 - 9.

[16] TCAS. 2006 [cited 2006 20th - October]; Available from: http://en.wikipedia.org/wiki/TCAS

[17] Hutchinson H, Qinetiq. ACAS II on Helicopters. *EUROCONTROL*. Brussels: European Organization for the Safety of Air Navigation 2006.

[18] Wicker R. Traffic Alert/Collision Avoidance System. 2006 29th - October - 2006 [cited 2007 13th - December]; Available from: http://www.allstar.fiu.edu/AERO/TCAS.htm

[19] Morring F. News Breaks - UAV Air Traffic Control. Aviation Week & Space Technology. 2006 6th - November 19 - 20.

[20] Editorial Staff. EUROCONTROL specifications for the use of military Unmanned Aerial Vehicles as operation air traffic outside segregated airspace. Eurocontrol 2007:8-11.

[21] Strong MCG. EUROCONTROL specifications for the use of military Unmanned Aerial Vehicles as operation air traffic outside segregated airspace. Eurocontrol 2007:8-11.

[22] Flight Safety Digest. See Whats Sharing Your Airspace. Flight Safety Digest. 2005 May 2005:1 - 26

[23] Grilley DE. Resolution Requirements for Passive Sense & Avoid. Morgantown, WV: Alion Science & Technology; 2005 26th - January.

[24] NASA ERAST. Non-Cooperative DSA Flight Test. *AUVSI 2003*. Baltimore, MD: NASA 2003.

[25] Panicker RA, Nigam N, Rajnarayan D. CS229 Project Report - Aircraft Collision Avoidance; 2005 16th - December 2005.

[26] Defence Science Board. Unmanned Aerial Vehicles and Uninhabited Combat Aerial Vehicles. Washington, D.C.: US Department of Defence; 2004 February.

[27] F38 Committee. S&A, Terminology, Launchers, VRF Practices, Commercial Pilot Practices, Manufacturing QA Practice. ASTM International F38 Meeting on Unmanned Aircraft Systems; 2006 13th - 14th November; Hyatt Regency Atlanta, Atlanta, Georgia USA: ASTM International; 2006. p. 1 - 2.

[28] Glover W, Lygeros J. A Multi-Aircraft Model for Conflict Detection and Resolution Algorithm Evaluation. Report to the IST European Commission: The Department of Engineering, University of Cambridge, UK and the Department of Electrical and Computer Engineering, University of Patras, Greece; 2004 18th February

[29] Prouty RW. Military Helicopter Design Technology. Malabar, Florida: Krieger Publishing Company 1998.

[30] Prouty RW. Even More Helicopter Aerodynamics. Maryland: Rotor & Wing 1993.

[31] Padfield GD, Lee DN, Bradley R. How Do Helicopter Pilots know When to Stop, Turn, or Pull Up? *American Helicopter Society 57th Annual Forum*. Washington, D.C. 2001.

[32] Bernier R, Bissonnette M, Poitevin P. DSA Radar – Development Report. *AUVSI 2005*. Baltimore USA 2005.

[33] AOPA Air Safety Foundation. Collision Avoidance - Strategies and Tactics. Safety Advisor - Operations and Proficiency.4.

[34] Flight Safety Technologies. Project Unicorn. 2007 [cited 2007 14th - April]; Available from: www.flysafetech.com

[35] Diefes D, Deering R. Concept Design for a Low Cockpit Display/Collision Avoidance System for General Aviation Aircraft. *Position Location & Navigation Symposium*: IEEE 1996:352 - 9.

[36] Scheiwzer Aircraft. Technical Data Summary: model 300C. Schweizer Aircraft 2005:Technical and Specifications Document for the Model 300C Helicopter.

[37] Robinson Helicopter Company. Robinson R44 - Raven I specifications. 2004 [cited 2007 21st - January]; Available from: www.robinsonheli.com/r44specs.html

[38] Zamora AG. VISIONAIR SW for Helicopters Operator's Manual. 3 ed: UAVNavigation 2007.

[39] Wall R. Managing UAVs in Europe. Aviation Week. 2007 10th December:34.

[40] Anderson DF, Eberhardt S. Understanding Flight: McGraw-Hill 2001.

[41] Karhoff BC, Limb JI, Oravsky SW, Shephard AD. Eyes in the Domestic Sky: An Assessment of Sense and Avoid Technology for the Army's "Warrior" Unmanned Aerial Vehicle. New York 2006.

[42] Larson J, Bruch M, Ebken J. Autonomous navigation and obstacle avoidance for Unmanned Survace Vehicles. San Diego: Space and Naval Warfare Systems Centre, San Diego (SSC San Diego) - US; 2006 April.

[43] Edrich M. Ultra-lightweight synthetic aperture radar based on a 35GHz FMCW sensor concept and online raw data transmission. IEEE Radar, Sonar, and Navigation. 2006 13th - April;153(2):129-34.

[44] Wolfe RC. NASA ERAST Non-Cooperative DSA Flight Test: Modern Technologies Solutions Inc.; 2003 31st - March to 4th -April.

[45] Ebdon MD, Regan J. Sense-and-Avoid Requirements for Remotely Operated Aircraft (ROA). In: Wells CLL, USAF, eds.: US Airforce - Department of Requirements 2004.

[46] Kermode AC. Mechanics of Flight. 9th ed. Essex: Longman Scientific & Technical 1991.