UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

MASTER THESIS

Design of a FMS to support four-dimensional trajectories

Manuel Angel Amaro Carmona

SUPERVISED BY

Dra. Cristina Barrado
Dr. Darius Rudinskas

Universitat Politècnica de Catalunya
Master in Aerospace Science & Technology

March 2014

This Page Intentionally Left Blank

Design of a FMS to support four-dimensional trajectories

BY

Manuel Angel Amaro Carmona

DIPLOMA THESIS FOR DEGREE

Master in Aerospace Science and Technology

AT

Universitat Politècnica de Catalunya

SUPERVISED BY:

Dra. Cristina Barrado
Departament of Computer Architecture - UPC, Barcelona, Spain

Dr. Darius Rudinskas
Antano Gustaicio Aviacijos Institutas (AGAI) - VGTU, Vilnius, Lithuania

ABSTRACT

The increase of air traffic rates registered in the recent years and the forecast for the next twenty years, has encouraged the necessity of developing new alternatives to manage the airspace. For this reason, several European and American organizations have been studying the idea of introducing new Air Traffic Management (ATM) systems based on four-dimensional trajectories.

The four-dimensional trajectory management consists in establishing far in advance a time-based sequence for all aircraft converging to a specific point. The main idea is providing each aircraft with a time constraint to get a specific merging point. As a result, it is allowed to the aircraft to perform autonomous flight in order to achieve this merging point in the required time.

In case the aircraft is not able to arrive to the merging point at the required time, it is assigned a new time of arrival; this means the aircraft has to perform Holding Patterns (HP) procedures, which is traduced to more fuel consumed, more emissions produced and several issues related to airlines and airport delays.

A new generation of avionics systems related to navigation and flight management is emerging in order to support precision trajectories based on waypoints composed by latitude, longitude, altitude and a fourth dimension represented by the time.

The purpose of this project is to design, simulate and test the functions of a Flight Management System (FMS) in order to follow automatically four-dimensional trajectories. This is achieved by controlling the aircraft airspeed, altitude, heading and vertical speed in order to arrive to the merging point in the specified time. The system receives data from the aircraft and computes new control parameters based on mathematical equations and prediction trajectories algorithms.

Additional features has been added to the FMS-4D, such as the capability of predicting the arrival time taking into account previous flight parameters and speed/altitude constrains.

Finally, a testing phase is performed using a flight simulator in order to obtain the performance and results of the designed system.

This Page Intentionally Left Blank

Table of Contents

List of Figures

List of tables

This Page Intentionally Left Blank

INTRODUCTION

The procedures and tasks related to the air traffic management have been improved since several years ago in order to increase the safety and reliability of the air transportation. The concept of four dimension airspace management has become more prominent in the recent years due to labors performed by the international organizations with the objective of taking the airspace into a new dimension of performance and reliability.

In this project, it is described the design of a Flight Management System (FMS) in order to support navigation procedures using four dimensional trajectories as part of the improvements for the future aircraft avionics and navigation systems necessary to evolution the air traffic transportation.

In the sections below the motivation and the specific objectives for this thesis are described as well as an outline to this report.

1.1. Motivation

According to Eurocontrol, the most-likely scenario C (Regulated Growth) has 14.4 million flights in 2035. This is equivalent to 50% more than 2012 and a growth average of 1.8% per year. This forecast pretends to alert to the international organizations to perform important improvements in the airspace management. In this scope, a possible solution in order to improve the air traffic transportation aims to the concept of 4D navigation. [1]

The 4D trajectory management consists in establishing far in advance a sequence for all aircraft converging to a specific point in a congested area. This is achieved using trajectory predictions computed by air traffic management ground and airborne systems. The main idea is providing each aircraft with a time constraint to get a specific merging point while allowing this aircraft to perform an autonomous flight in order to achieve this merging point in the given time. [2]

If the merging point is not achieved in the specified time, the aircraft has to perform holding patterns in order to wait a new time to achieve the merging point and to avoid disrupting the flight sequence predicted previously by the ground systems. Several problems are related to this fact; in one hand, the aircraft consumes more quantity of fuel which is traduced to economic losses for the airline companies and even worst the CO_2 emissions produced by this consumption. In the other hand, a new merging point sequence has to be computed by the ground systems and this produces delays to the airlines as well as conflicts in the airports.

According to the problems described above, it is important to provide the aircraft with more accurate avionics systems capable to follow precise trajectories to achieve the specific merging point in the assigned time with a very low errors tolerance.

1.2. Thesis Definition

The Flight Management System (FMS) provides excellent capabilities for navigation purposes. It provides flight control of the aircraft in order to navigate according to a given (and managed) flight plan. The next generation of these avionics systems aims to support four dimensional trajectories.

The aim of this thesis is designing an improved FMS in order to follow four dimensional trajectories with high precision based on a given flight plan with time constraints.

In order to achieve this, it has been proposed the following more specific objectives:

- Studying the state of the art of aircraft navigation based on four-dimensional trajectories and its application to the Air Traffic Management Systems.
- Designing the algorithms and functions of an improved FMS in order to support four dimensional trajectories.
- Implementing the functions of the designed FMS-4D into a software application.
- Testing the FMS-4D functions using a virtual flight simulator.

1.3. Report Outline

This report is composed by the following chapters:

Introduction: it provides a general overview of the thesis showing the most important elements as well as the motivation and the objectives set.

State of the Art: this chapter provides an overview of the current projects and research related with the aircraft navigation based on four dimensional trajectories and its application to the Air Traffic Management (ATM) systems.

Theory: It provides the concepts related with the design of the Flight Management Systems as well as concepts related to the navigation system

FMS-4D Design: This chapter describes the software and hardware design of the improved Flight Management System in order to support four-dimensional trajectories.

Test and Results: it describes the test performed to the system as well as the results and the performance of the designed system.

Conclusions: summarizes the work performed as well as gives an insight of the future work related to this project.

This Page Intentionally Left Blank

Chapter 2

STATE OF THE ART

The 4D navigation is considered one of the most essential features of the future Air Traffic Management (ATM) systems. This chapter provides an overview of the European and American programmes designed to improve the current airspace in order to be a safer and environmentally enhanced place. In addition, it is shown the ultimate information regarding to the most recent avionics systems used to support navigation in four dimensions.

2.1. Single European Sky ATM Research (SESAR)

In 2004, the European Commission launched the initiative to restructure the European airspace according to the air traffic in order to create additional capacity and increasing the overall efficiency of the current ATM systems. For this reason it was created the Single European Sky ATM Research (SESAR). [3]

According to SESAR [3], it is developed a new air traffic management system in order to enhance the capabilities of the European airspace. This project is developed in three phases.

The first phase was developed between 2004 and 2008 and it was aimed to define the development, deployment and planning for the next generation of air traffic management systems.

The second phase, from 2008 to 2013, is aimed to produce this new generation of technological systems as well as all the operational procedures developed in the first phase.

Finally, the third phase is aimed to the implementation and the large scale production of the components of the new project. This phase is expected to be performed from 2014 to 2020.

2.2. Next Generation Air Transportation System (NextGen)

The NextGen is new version of the National Airspace System (NAS) of the United States. This system aims to update the current NAS in the period between 2012 and 2025. [4]

Some of the new features of the systems are that the airplanes will be able to flight closer together, take more direct routes, saving time, reduce fuel consumption, reduce traffic delays and increase the capacity.

In NextGen, the use of four-dimensional trajectories is studied in order to define an accurate description of the aircraft path in space and time. The waypoints are associated with the path as specific steps along it. These waypoints could be associated with a Controller Time of Arrival (CTA) which has specific time tolerances [5].

2.3. Controlled Time-of-Arrival (CTA) Flight Trials

The necessity of utilizing four-dimensional navigation procedures in the enhanced air traffic management systems has boosted the efforts to update the new avionics systems in order to accomplish required time of arrival functions in a more efficient way. For this reason, in matter of research about 4DT, some simulations and flight trials have been performed using enhanced Flight Managements Systems (FMS).

This section describes the objectives of the flight trials and shows their most important results.

2.3.1. SESAR's "Initial 4D" Flight Trial and Simulations

In February 2012, the SESAR programme performed the "Initial-4D" flight. An Airbus A320 test aircraft flew from Toulouse to Stockholm using for the first time an I-4D flight profile. In the four-dimensional flight, the aircraft and the air traffic management systems exchanged data related to four dimensional trajectories using a data link [2].

The benefits regarding to this programme turn around the possibility of performing a safer and predictable flight which is traduced to a more efficient planning and an increase of the capacity of the airports and the European airspace.

In a research project [6] conducted by SESAR, a one-week simulation was performed where eight pilots of European airlines were involved in order to test the behavior of the crew and the aircraft when flying using time constraints in a four dimensional navigation environment.

The simulation was taken in a subset of the Maastrich airspace, where it is known as a high air traffic area. The area covers the waypoints DELTA, RUHR and MUNSTER. The aircraft started in cruise level and included the phase from initial level to a minimum FL260. The descend lasted approximately 40 minutes where the crew had to arrive to a merging waypoint in a Controlled Time of Arrival (CTA) with a tolerance no more than ±30 seconds.

The simulation was performed in an Airbus A320 simulator equipped with an emulation of a Honeywell Required Time of Arrival (RTA) FMS function. The pilots were instructed to respect the time constraints provided by the ground systems by using the RTA function. The task of arriving to the waypoint in the specific time of arrival was left to the pilots who could use airspeed control or other maneuvers.

Figure 2.1: RTA function, in selected (left) and managed (right) speed modes [6]

As result of the simulation, the pilots could follow the 4D trajectories on cruise level by gaining or losing time in order to respect the RBT time[1] with no difficulty. However, for the CTA tolerance, the challenge was different and the pilots and controllers had to coordinate in order to achieve the required time constraints with the required tolerance. The preference of the pilots was to perform speed correction instead of lateral maneuvers.

According to the report provided by SESAR [6], "*Improvements to the current RTA functions are required. The computation of the managed speed as well as estimated times of arrival should be enhanced to increase the robustness of the guidance, especially during descent. Moreover, additional support (late/early indication on the navigation display, and a "what if" tool) were requested for monitoring and selecting appropriate action.*"

These results show that it is necessary to improve the Flight Management Systems (FMS) in order to support better guidance of the aircraft while following four-dimensional trajectories.

2.3.2. CASSIS and SAS Flight Trials

From September 22 to 25, 2008, the project CASSIS has conducted a set of flight with SAS where the data collected was used to study the suitability of the use of a FMS-4D trajectory and time-control behavior [5].

The experiment was conducted in a Boeing B737-600 and B737-800 of SAS provided by an on-board Flight Management System (FMS) with the GE Aviation System Update 10.7 and some implementation of ARINC 702A-1 trajectory bus outputs in order to share in networks parameter of interest of the flight. The information was sent using the data-link of the SAS ACARS.

In all the flights, the aircraft used the same FMS model which will be provided with a CTA function by the manufacturer.

---

[1] *It is the target time that relies on the concept of reference business trajectory (RBT) which the aircraft should fly and the service provider facilitate. This time have a tolerance of -2 minutes and +3 minutes. Sometimes this time is not enough in order to regulate the traffic flow sequence. For this reason, in busy airports it is taken the Controller Arrival Time (CTA) instead.*

After reviewing the information and discarding the flights that were affected by certain anomalies such as ATC parameters or vector changes, the results show that the mean error when complying the CTA Time at the Initial Approach Fix (IAF) was around 4 seconds. In some flights, the time constraint was placed in the runway threshold instead, which is located to 19 NM of the IAF. For these cases the mean accuracy obtained was of 15 seconds [7].

After analyzing these results, the study shows that an important factor of affecting the CTA behavior is the accuracy of the wind estimation used by the FMS to predict the estimated time of arrival (ETA).

Also, some CTA errors are related with the use of some elements of the aircraft. In this case, in [7] it is detailed that the earlier flap extensions are correlated to late CTA errors. Conversely, the later extensions coincide with early CTA errors during the landing phase.

2.4. Flight Management Systems for 4D Navigation

The concept of four dimensional trajectories incorporating the temporal dimension into operations cannot exist without accurate time of arrival capabilities that allow the aircraft to satisfy the required time constraints. These capabilities are achieved using Flight Management Systems (FMS) with more advanced features.

In this section, it is described the most common Flight Management Systems (FMS) supporting four dimensional navigation.

2.4.1. Next Generation Flight Management System (NG-FMS)

The Next Generation Flight Management System or NG-FMS is an avionic system developed and launched in 2010 by Honeywell© Inc. The system is based on performance that meets both the Single European Sky ATM Research (SESAR) and the NextGen Air Traffic Management (ATM) objectives [8].

According to a research performed by *R. Sabatini et al.* [9], the Next Generation Flight Management System (FMS) generates optimal trajectories (operationally and environmentally) thanks to mathematical models which were developed for this purpose. In the performance field, several cost functions have been considered in order to optimize the fuel consumption, the flight time and the CO_2 emissions. In addition, the models include aircraft dynamics, engine, atmospheric, noise and weather.

Other important features of the NG-FMS are enabling the Required Navigation Performance (RNP) 0.1, support Wide Area Augmentation System – Localizer Performance with Vertical guidance (WAAS-LPV), Future Air Navigation System 1 (FANS-1) and FANS-2. [8]

This system is currently used by few modern aircraft like the Boeing 747-8 as a stand-alone system and integrated in other avionics systems of the Gulfstream G650.

2.4.2. Flight Management Computer (FMC) and 10.7 GE Aviation Systems Update

The Flight Management Systems used in the current new generation of Boeing B737-NG (-600 / 700 / 800) are provided with a Trajectory tracking function, which is a feature enabled by the 10.7 GE Aviation Systems Update [7].

These FMS use ARINC 702A-1 Trajectory Bus which is a technology capable to send each minute or when the Flight Plan (FP) changes, information about the state of the aircraft. This is latitude, longitude, altitude, turn radius, turn direction, fly-by waypoint type, and of course time. With this information, it is possible to obtain the full aircraft trajectory including vertical waypoints and turns.

The main idea of the use of this bus is to track this information and sending it through an ACARS download data-link or output on a dedicated ARINC 429 in order to perform four-dimensional trajectory intent.

The first trial flights utilizing this technology were performed in Sweden with the collaboration of SAS Airlines [7].

Chapter 3

THEORY

This chapter provides the concepts necessary to understand the design of the Flight Management System (FMS) described in the next chapter. Initially, it is given a briefly introduction of the concept of four dimensional navigation, subsequently it is boarded the mathematical background of the trajectory parameters estimation and finally it is explained a number of concepts concerning to the designed system, as well as the communication protocol used to associate the system with the simulation environment.

3.1. Four Dimensional Navigation

In order to perform navigation, it is necessary to know the initial waypoint and the final waypoint (and are provided in the flight plan). In the mathematical scope, a point in the space is composed by three dimensional components (x, y and z). However, when these points are used in order to represent a waypoint, it is taken its position with respect to the Earth; this position is given by its geographical coordinates.

Hence, a three dimension waypoint is composed by latitude, longitude and altitude.

The four dimensional navigation consists in achieving a three dimensions waypoint at a required time of arrival by changing the aircraft flight profile. In this context there are several variables and parameters involved in the concept of changing a flight profile. According to *S. Mohleji* [10], the airspeed is considered the most important control variable in order to achieve a waypoint at a given time of arrival.

In accordance with the information provided above, a waypoint is defined by the components shown in the Table 3.1. It is important to keep in mind this information in order to understand the following sections and chapters of this project.

Table 3.1: Parameters identification

Parameter	Provided by	Description
Latitude	Flight Plan (ATM)	Describes the north-south position of a point on the Earth's surface
Longitude	Flight Plan (ATM)	Describes the east-west position of a point on the Earth's surface with respect the Greenwich meridian.
Altitude	Flight Plan (ATM)	Describes the position of a point with respect to the Sea Level
Time of Arrival	ATM Ground Systems	Describe the time provided by the ATM ground systems when the aircraft should arrive to a specific waypoint.

The air traffic management systems administer the airspace in order to create a sequence of aircraft that expect to arrive to a merging point. Consequently the aircraft are sequenced in a time-based list thus each aircraft is provided with a time constraint or a required time of arrival. In addition, each aircraft is instructed to intercept the merging point at the assigned time.

The Figure 3.1 shows the typical sequence of numerous aircraft around an airport to be administered by the traffic management systems.

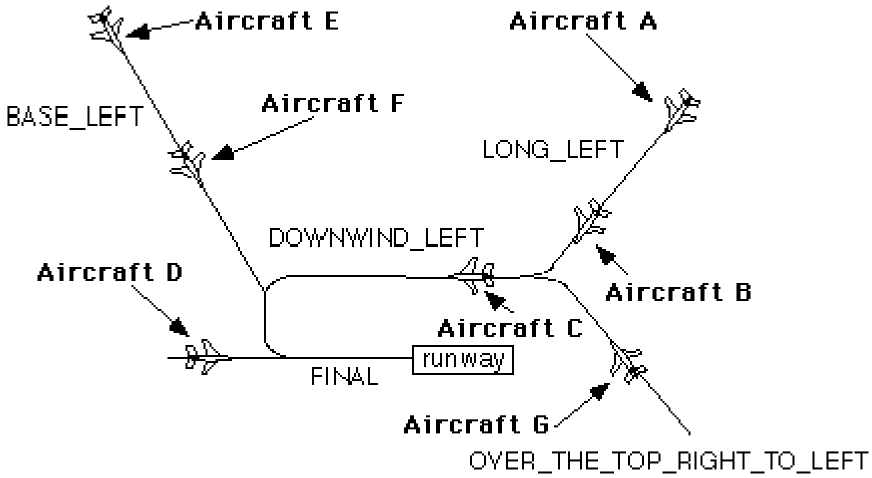

Figure 3.1: Sequencing Example [11]

As shown above, the aircraft arrive to different merging points (e.g. Initial Approach Fix (IAF)) before performing the final approach to the runway. Each aircraft is listed to the sequence by assigning a time constraint, which is equivalent to a specific time of arrival. Consequently, each pilot has to perform the relevant changes in the aircraft control systems in order to match this time.

3.2. Trajectory Parameters Estimation

A trajectory can be defined as the path followed by an object moving under the action of given forces. According to this definition, when an aircraft moves from a three dimensions waypoint composed by latitude, longitude and altitude to another point with similar components, it is possible to affirm that the aircraft is "*following a trajectory*".

When an aircraft is following a trajectory, it is possible to obtain some parameters that define this action. For example, it is possible to measure the distance flew between the two points, or the heading followed by the aircraft to achieve the final point or even the flight average airspeed and the time used to pursue this trajectory.

In the same way that it is possible to track this information, the trajectory prediction concept of this project is used reversely in order to estimate the behavior of these parameters, by departing from the statement that the initial point and the final point of the trajectory are known. This is achieved using mathematical modeling and equations that are described in the following subchapters.

3.2.1. Parameters and Variables Identification

In order to understand the theory of the following subchapters regarding to the mathematical equations for navigation, it has been decided to summarize the following general parameters and variables into the Table 3.2.

Table 3.2: Parameters used in mathematical equations

Parameters	Details
$\emptyset$	Longitude in radians of a given point
Λ	Latitude in radians of a given point
$\Delta\lambda = \lambda_1 - \lambda_2$	Difference between the latitudes (in radians) of the two points.
$\Delta\emptyset = \emptyset_1 - \emptyset_2$	Angular distance in radians

3.2.2. Distance Estimation

The distance estimation is considered one of the most important parameters of the mathematical model for navigation and trajectory prediction. The Haversine equation for orthodromic distances provides simplicity for computational calculation and it is used for navigation purposes.

The equation provides the orthodromic distance between two points in a sphere. As mentioned above, this equation is often used for navigation purposes due to its great precision for small distances [12], because of this, it result to be used for estimation of four-dimensional precision trajectories.

Table 3.3: Constant and function used in Haversine equation

Element	Details
$atan2$	Arctangent functions with two arguments (computational function)
$R = 6371\ kms$	Earth Radius

In order to understand it easily, the Haversine equation [12] can be divided in three main elements.

The square of half the chord length between two points denoted by:

$$a = sin^2\left(\frac{\Delta\emptyset}{2}\right) + \cos(\emptyset_1) . \cos(\emptyset_2) . sin^2(\frac{\Delta\lambda}{2}) \tag{3.1}$$

The angular distance expressed (in radians) by:

$$c = 2. atan2(\sqrt{a}, \sqrt{1-a}) \tag{3.2}$$

The orthodromic distance expressed (in kilometers) by:

$$d = R.c \tag{3.3}$$

3.2.3. Bearing Estimation

The bearing refers to the direction that the aircraft have to move in order to reach final point of a path. The bearing could change depending of the latitude of the initial and final point as well as the total distance of the path. Because of this, it is possible to compute an *Initial Bearing* and a *Final Bearing*.

However, when the distance between two points is really small, the Initial Bearing value is closer to the Final Bearing value. [13]

The equation of the initial bearing from a given initial point to a final point in a straight line along a great-circle arc is denoted by the following equation:

$$\theta = atan2(\sin(\Delta\lambda) . \cos(\emptyset_2) , \cos(\emptyset_1) . \sin(\emptyset_2) - \sin(\emptyset_1) . \cos(\emptyset_2) . \cos(\Delta\lambda)) \tag{3.4}$$

3.2.4. Time Estimation

The time that an aircraft spend to move from an initial point (*lat*, *lon*) to a final point (*lat*, *lon*) can be represented as following:

$$t = \frac{1}{\beta} \cdot \int_{x1}^{x2} \frac{\partial x}{V_a(z_c)} \tag{3.5}$$

where V_a is the true airspeed of the aircraft at a given altitude and β is a conversion factor from *knots* to *feet / seconds.*

3.2.4.1. Wind Effect

An important element that has to be taken into account for time computation is the along-track wind effect. The wind produces an important change of the airspeed of the aircraft and this could affect the estimation of the time of arrival. The effect of the along-track wind over the aircraft airspeed in cruise level is associated with two elements [10]:

1. The direction of the wind with respect to the aircraft. Depending of the relationship between the heading of the along-track wind and the aircraft, it is called tailwind or headwind.

2. The magnitude of the along-track wind which represents the constant velocity of the wind at a given altitude (cruise altitude).

Table 3.4: Constants and variables of the wind effect equations

Constants and Variables	Details
V_w	Speed of the wind at the given altitude.
H_w	Heading of the wind at the given altitude.
δ	Factor to correct the magnetic north to the true north.
B	Bearing of the aircraft track.

The along-track component of wind in the horizontal plane can be represented as following [10]:

$$W(z_c) = V_w(z_c) \cdot \cos\left(|B(z_c) - H_w(z_c) \pm \delta|\right) \tag{3.6}$$ [2]

By using the equation 3.5 into 3.6, the resultant equation to compute the time an aircraft flight between an initial point and a final point with constant velocity, is the following:

$$t = \frac{1}{\beta} \cdot \int_{x1}^{x2} \frac{\partial x}{V_a(z_c) + \left[V_w(z_c) \cdot \cos\left(|B(z_c) - H_w(z_c) \pm \delta|\right)\right]} \tag{3.7}$$

By integrating the equation 3.7, the result is the time estimation equation with wind effect.

---

[2] *The along-track component of the headwinds is negative and for tailwinds it is positive.*

$$t = \frac{1}{\beta} \cdot \frac{[x_{2-}x_1]}{V_a(z_c) \pm [V_w(z_c).\cos(|B(z_c) - H_w(z_c) \pm \delta|)]} \tag{3.8}$$

Note that this equation can be used in order to estimate the minimum time at which an aircraft can fly from the initial point to the final point at cruise airspeed. In addition, it is possible to isolate the airspeed in order to obtain the velocity that an aircraft should fly in order to arrive to the final point in a specific time or even it is possible to isolate the factor $[x_{2-}x_1]$ in order to estimate the distance flew in a specific time at a given velocity.

This is an important statement that should be remembered in order to understand the computation of the flight control parameters explained in the next chapter.

3.2.5. Vertical Speed Estimation

The vertical speed is the rate of climb or descent of an aircraft. It can be estimated using the following equation:

$$VS = \frac{|Z_1 - Z_2|}{t} \cdot \frac{1}{\partial} \tag{3.9}$$

Where ∂ is a conversion factor from seconds to minutes and $Z_1 - Z_2$ is the difference of altitude between the initial point and the final point of the vertical trajectory.

The vertical speed is represented as a negative value when the aircraft is descending and positive value for the contrary case.

3.3. Universal Datagram Protocol (UDP)

It is a protocol commonly used in network communications. The protocol is based in datagrams that provide information referenced in a heading tag. [14]

A typical X-Plane datagram is composed by 41 bits. The first four (4) bits represent the word "*DATA*" which means that the datagram is composed by useful information for the flight simulator. The bit number five (5) is just a *check* bit. The next four bits from positions six (6) to nine (9) are the "*Data Set*". This means to which parameter or group of parameters belongs the data. The next 32 bits represent the data to be sent or received.

A typical UDP datagram is showed as following

```
Byte = {
‘DATA’      68, 65, 84, 65,
‘Check’     0,
DataSet     11, 0, 0, 0,
Value 1     205, 204, 76, 62,
Value 2     0, 192, 121, 196,
Value 3     0, 192, 121, 196,
Value 4     0, 192, 121, 196,
Value 5     0, 0, 0, 0,
Value 6     0, 0, 0, 0,
Value 7     0, 0, 0, 0,
Value 8     0, 0, 0, 0
}
```

For some datasets there are useful less than 32 bits, because of this is possible to see zero-padding effects in several datagrams (as shown in the example given above).

Despite of several disadvantages of this communication protocol related to its reliability for critical operations and real-time systems, the UDP protocol is used by the Flight Simulator X-Plane in order to send and receive information about the data of the simulation. [15]

3.4. Flight Management System (FMS)

The Flight Management System or FMS is an avionic system commonly used in commercial and military aircraft. The main function of a FMS is performing in-flight tasks related to the flight plan. The FMS uses GNSS information in order to track the aircraft position and guiding it along the flight plan.

In the hardware scope, the FMS is controlled by the crew through a Control Display Unit (CDU) composed by a display and several function keys. Also, it interacts with several avionics elements such as the Primary Flight, Navigation, Multifunction and Engine displays, flight control system, engine and fuel system, data link system and surveillance systems [16].

Figure 3.2 Control Display Unit (CDU)

In the software scope, the FMS is composed by a navigation database of elements used for compose a flight plan and commonly defined by the ARINC 424 standard. The information provided by the database includes waypoints, airways, aids, NDBs, VORs, airports, Instrument Landing System (ILS), Standard Instrument Departure (SID), Standard Terminal Arrival (STAR), Holding Patterns (HP) and Instrument Approach Procedures (IAP).

The FMS software core is composed by precision trajectory algorithms used to compute flight distances, time of arrival, fuel consumption, Top of Descent Points (TOD), headings, optimal rate of climb/descent, and other values related to the flight plan.

The FMS support navigation in the lateral (LNAV) and vertical plane (VNAV).

In this case, the Lateral Navigation (LNAV) refers to the control of different parameters (heading) in order to navigate from point to point in the horizontal plane. This navigation uses geographic coordinates in order to compute the path from a waypoint to the next one.

In the other hand, the Vertical Navigation (VNAV) refers to control the different parameters (altitude and vertical speed) in order to navigate from point to point in the vertical plane.

For both, lateral and vertical navigation, the FMS is capable to follow some rules denoted as constraints of the flight plan and / or airspace international regulations. E.g.: FL110 / 250 IAS. (This means that below 11.000 feet the airplane cannot increase beyond 250 knots) [17].

Chapter 4

FMS-4D Design

This chapter describes the design of an enhanced Flight Management System (FMS) with features that support navigating by using four dimension trajectories. The algorithms and software described in this section are considered the core of the solution provided by this thesis.

4.1. Algorithms and Software Design

The software of the FMS-4D is composed by a user interface which represent the control display unit of the FMS, a database system that provides information relative to the navigation waypoints, the communication interface that links the software with the flight simulation X-Plane and the control system which uses the trajectory prediction equations in order to compute control parameters to perform four-dimensional navigation.

4.1.1. User Interface Design

As mentioned above, the User Interface (UI) represents the Control Display Unit (CDU) of the Flight Management System (FMS-4D). It is composed by a display that shows a feedback between the pilot and the system. The function buttons, the directional, character and numeric pads and the flight plan input buttons.

More information about the buttons shown in this Control Display Unit (CDU) is provided in the user manual as annexes information to this report.

Figure 4.1: User Interface (UI)

4.1.2. Required Properties and Input / Output Data

The algorithms and software designed in this thesis are based on the mathematical models explained in the previous chapter, control techniques and a communication interface with a flight simulator. Because of this, it is important to determine a data format and a proper communication interface.

4.1.2.1. Data Format

The data format is used to represent the information that is managed by the system.

The elements "*Flight Plan*" and "*Flight Plan Entries*" represent the information regarding to the path followed by the aircraft. These elements are used as the input data of the control system.

In the other hand, the "*UDP_Pack*" represent the information used by the communication interface in order to send / receive information to / from the flight simulator.

These elements are described in detail as following:

Flight Plan Entry (Structure): Is the structure which contains the information relative to an entry of the flight plan. This could be an airport (APT), NDB, VOR-DME (VOR) or intersection (FIX). In the same way, it is composed by the geographic coordinates, constraints and ID used in the navigation. The Table 4.1 shows this data format.

Table 4.1: Definition of the Flight Plan Entry Data Format

Data Type	Type	Description	Unit
string	type	Type of entry. Airport, NDB, VOR or Intersection	-
string	name	Name of the waypoint	-
double	lat	Latitude	Degrees
double	lon	Longitude	Degrees
double	alt	Altitude	Feet
double	time	Time of Arrival from the previous point	Seconds
double	hdg	Heading	Degrees
double	spd	Speed for constraint purposes	Knots
double	fuel	Fuel from previous point	Galons
string	id	IATA identification	-

Flight Plan (Class): Is the class composed by elements of the type "Flight Plan Entry" (defined by default quantity 200.) The class contains the typical *add*, *edit* and *remove* methods as well as elements for description of the flight plan. The Table 4.2 shows this data format.

Table 4.2: Definition of the Flight Plan Data Format

Data Type	Type	Description
Flight_Plan_Entry	list	Vector of entries. (default number 200)
int	iter	Iterator
string	route	Route of the Flight Plan
string	description	Description of the Flight Plan

UDP Pack (Class): This class provides an interface of communication between the Control System and the Flight Simulator using the UDP standard format. It is composed by methods such as *Send (UDP_Pack Pack)* and *Receive()* for treat accordingly the information. The Table 4.3 shows this data format.

Table 4.3: Definition of the *UDP_Pack* Data Format

Data Type	Type	Description	Unit	Operation
double	Auto_HDG	Autopilot Heading	degree	Send
double	Auto_ALT	Autopilot Altitude	feet	
double	Auto_SPD	Autopilot AirSpeed	knots	
double	Auto_VS	Autopilot Vertical	feet / min	
double	Curr_LAT	Current Latitude	degree	Receive
double	Curr_LON	Current Longitude	degree	
double	Curr_ALT	Current Altitude	feet	
double	Curr_HDG	Current heading	degree	
double	Curr_TSPD	Current true airspeed	speed	
double	Curr_SPD	Current airspeed	knots	

4.1.2.2. Database Properties and Structure

The project of exporting navigation data created by Megginson Technologies® Ltd. provides some *.csv* files (updated on October 2013) with valuable information about AIDS, Airports, Waypoints and Navigation Data [18]

This information has been processed in order to create a database from it. The database format has been chosen as SQLite because is a compressed format that can be used for applications with limited resources [19].

The structure of the database is really simply because it is used only for reading purposes. The structure of the database is shown in the following table.

Table 4.4: Database structure

Table	Description
airports	Composed by 44684 airports
fixes	Composed by 119721 waypoints
navaids	Composed by 11126 NDB and VOR

4.1.3. Navigation with time constraints

The navigation algorithms pretend to calculate flight control parameters of the aircraft based on a target waypoint composed by a latitude, longitude, altitude and time of arrival.

4.1.3.1. Navigation Parameters

In order to perform navigation with time constraints (4D) it is necessary to take into account some parameters that are used to define the target waypoint and the flight profiles of the aircraft while navigating.

These parameters are described in the following table.

Table 4.5: Navigation Parameters

Parameter	Description	Unit
Latitude	Defined in Flight Plan	Degrees
Longitude	Defined in Flight Plan	Degrees
Altitude	Defined in Flight Plan and could be changed depending of the weather conditions or ATM procedures.	Feet
Time of Arrival	Defined in the FMS Flight Plan according to the aircraft limitations and discussed with the crew.	Defined in seconds for this design, due to simplicity. But it can be a standard time.

4.1.3.2. Flight Control Parameters

According to the previous chapter, the equation 3.8 describes how to compute the time that spent an aircraft while flying a given distance at a given airspeed. Also, the equation 3.4 describes the bearing of the aircraft in order to achieve a point expressed in geographical coordinates.

Therefore, if the time of arrival is a constant value as well as the position and altitude of a target waypoint (because the ground systems provide the aircraft with this information), it is possible to use this information in order to compute a new set of parameters that let the aircraft achieve the waypoint at the required time of arrival. This set of parameters is explained in the sub-sections below.

4.1.3.2.1. Heading Control

The heading control allows the aircraft to perform a correct navigation in the horizontal plane. The basic idea is computing the heading necessary to achieve a target waypoint (composed by latitude and longitude) from the current aircraft position (composed by latitude and longitude as well).

To perform this control, the position of the aircraft is received and by using the equation 3.4 from the previous chapter, it is computed a new heading which is sent to the aircraft autopilot panel. The Figure 4.2 shows the Heading Control performed by the FMS-4D over the aircraft.

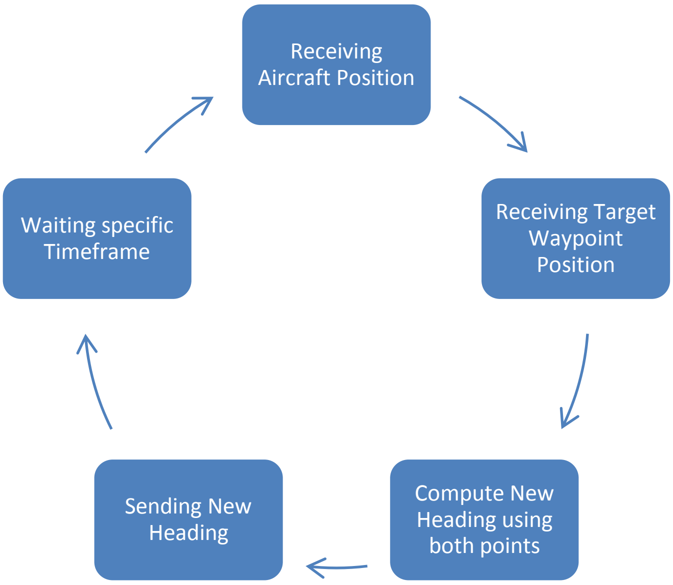

Figure 4.2: Heading Control

The heading control function allows the aircraft to go *directly* to a waypoint. To simplify things, it does not take into account a specific track-course or route. For this reason, the crew will have all the time the final decision about if using the autopilot function NAV[3] or HDG while navigating using the FMS-4D.

---

[3] *The NAV function provides to the crew a feature that allows the aircraft navigating to a VOR, NDB or a specific GPS waypoint following a course while adheres to a course-track.*

This would be suitable for the waypoints where specific navigation course functions are required, for example trying to intercept a VOR with a specific track-course. It is important to highlight that this action does not affect the other control functions of the system; simply the heading computed by the heading control would be ignored until the VOR is reached. Afterwards, it can be switched back the HDG function on.

4.1.3.2.2. Vertical Speed Control

In the same way that the aircraft uses the Heading Control in order to achieve a point in the horizontal plane, in the vertical plane, the situation is similar. However, in this case it is necessary to understand the behavior of the algorithm when there are speed / altitude constraints.

When the aircraft is flying in a constrained situation (e.g. below a constraint flight level), the vertical speed remains constant. This has been designed in order to obtain a correct prediction of the time when the aircraft crosses the constraint altitude. (*This behavior is boarded deeper in the sections 4.1.4.1* Speed and Altitude Constraints *and 4.1.4.2 Minimum Arrival Time Prediction*).

In the other hand, when the aircraft is flying in a non-constrained situation (e.g. over a constraint flight level), it is necessary to know a preliminary prediction of the time of arrival to the target point. By using this information, the FMS-4D computes the ascending or descending rate in order to achieve the required altitude in the exact time (or even before) when the aircraft arrives to the target waypoint.

If this is the case, the vertical speed is computed using the equation 3.9 and the vertical speed control is performed as shown in the following figure.

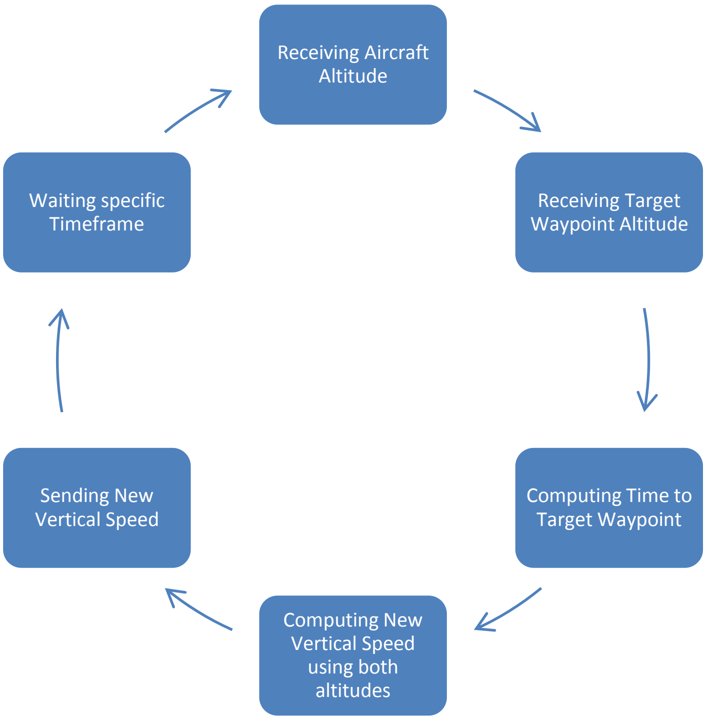

Figure 4.3: Vertical Speed Control

It is important to point out, that if the aircraft is flying in a constrained situation, such described above, the vertical speed control is performed simpler, this is using a default ascending / descending rate of the aircraft which is previously defined in the FMS-4D algorithm.

4.1.3.2.3. Airspeed Control

The airspeed control is considered the core of the flight control regarding to four-dimensional navigation. The airspeed is the most important parameter to validate the aircraft arrives to the target waypoint at the expected arrival time (*basically, the main goal of the four-dimensional navigation*).

The wind effect, as well as other factors related to weather conditions can disturb the proper performance of the aircraft, which is traduced into an eventual change of the airspeed. Because of this, the airspeed control has to take into account these random effects.

The airspeed of the aircraft is sensitive to different actions. When an aircraft is ascending, the airspeed tends to decrease (also in the other way around). In addition, the aircraft airspeed is modified by the flaps position, airbrakes and landing gear.

However, most of these changes can be controlled with a change in the aircraft thrust, which is traduced directly as a change of the aircraft airspeed.

The equation 3.8 describes the time spent by an aircraft while flying a known distance at a given airspeed. The main idea of the airspeed control is that the time and the distance are known parameters[4]; therefore it is possible to obtain the required true airspeed of the aircraft in order to arrive to the target waypoint in the defined arrival time.

The airspeed provided by the equation 3.8 refers to *true airspeed (TAS)*. However, the airspeed sent to the aircraft is indicated airspeed. Since the true airspeed and the airspeed are related with the density, their relation changes according to the altitude of the aircraft (because an eventually change of the density).

For this reason, the *true airspeed (TAS)* is converted to *indicated airspeed (IAS)* before sending to the aircraft using the density information provided by the variables of the simulator.

The following figure shows how the airspeed control is performed in the FMS-4D.

---

[4] *The time is computed continuously and this explained deeply into the section 4.1.5 Program Structure and Flow and the distance is computed using the Haversine equation (3.3)*

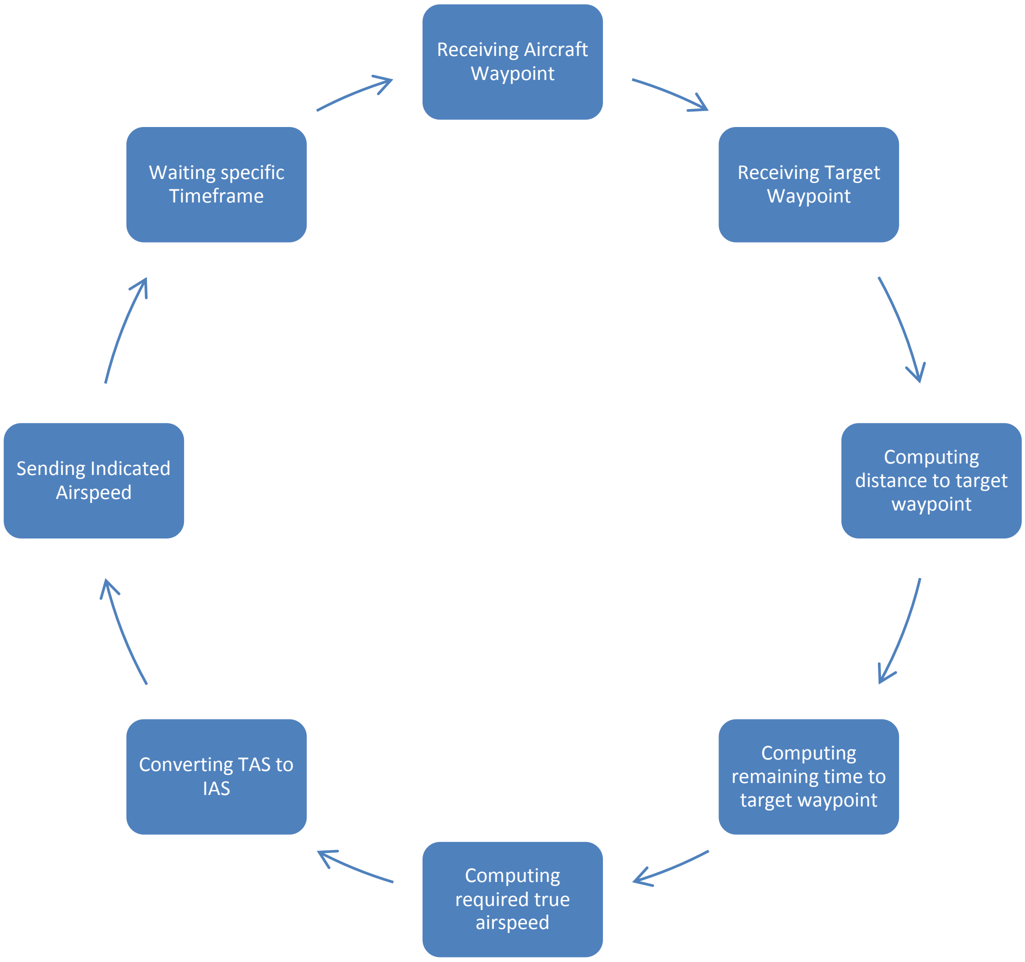

Figure 4.4: Airspeed Control

Additionally to the routines shown in the Figure 4.4, it has been added a proportional controller feature. The main idea is to improve the airspeed computation with respects to the delays performed by the aircraft (the airspeed changes slower than other parameters such as vertical speed and heading). The proportional controller consists in calculating the difference between the aircraft airspeed and the airspeed value calculated by the airspeed control function (error) each time-step and added *proportionally* to the airspeed control function output. In this case, the *Kp*[5] value has been set up empirically. After testing several values, the parameter *Kp* has been set to 90% of the computed error.

It is important to point out that, if the aircraft is flying in a constraint situation (e.g. below a constraint flight level), the airspeed remains constant until the aircraft reaches the constraint flight level.

This situation is taken into account to calculate the minimum *Predicted Arrival Time (PAT)* and it is explained deeper in the section *4.1.4.2 Minimum Arrival Time Prediction* below.

---

[5] *The Kp value represents the proportional gain of the controller.*

4.1.4. Important Features

4.1.4.1. Speed and Altitude Constraints

In the previous sections (*4.1.3.2.2* and *4.1.3.2.3*), it has been mentioned the existence of *speed and altitude constraints*; this section pretends to describe how it is managed the speed and altitude constraints in the FMS-4D as well as its relation with the flight control parameters.

According to the airspace for commercial aircraft, there are some limitations related to the speed of the aircraft below certain flight levels [17]. Because of this, it is important to design an FMS which takes into account such limitations.

These limitations take a direct effect in the airspeed control because, as mentioned above, the constraints are used to limit the airspeed of the aircraft that fly below a specific flight level. In addition, these speed limitations produce an important effect in the time at an aircraft is capable to achieve the target waypoint. This effect is boarded in the following section.

The main idea of the speed constraints is that the FMS-4D performs a *check* before computing a required airspeed. If the flight level is below than the constraint flight level, then the airspeed is not computed using the airspeed control loop but it remains constant at the constraint airspeed (except that the airspeed control loop determines that the airspeed have to be lower than the constraint airspeed).

The Figure 4.5 shows the typical flow performed by the airspeed control in order to follow the airspeed constraint.

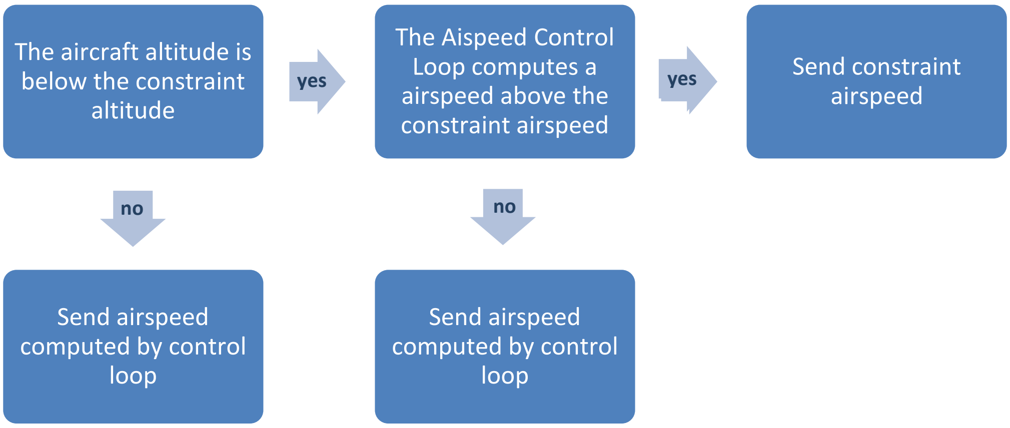

Figure 4.5: Speed / Altitude *Check* performed by airspeed control

4.1.4.2. Minimum Arrival Time Prediction

What could happen if the ATM Ground Systems assigns an arrival time that cannot be achieved by the aircraft due to technical limitations or due to speed / altitude constraints?

This section pretends to get a response to the question above by explaining one of the key features of the FMS-4D: How to compute the *Minimum Predicted Arrival Time (PAT)*.

The *Predicted Arrival Time* (*PAT*) is a parameter designed for this FMS-4D in order to give to the pilots an idea of the best performance of the aircraft. This parameter pretends to inform to the crew the minimum time that the aircraft requires in order to achieve a target waypoint. By using this information, the crew could discuss with the ATC if is possible to be assigned a specific arrival time or if this time has to be delayed instead.

When a new waypoint is added to the FMS-4D, a new PAT for that waypoint is computed taking into account the following parameters and / or situations:

- The maximum aircraft cruise speed.
- The existence of speed / altitude constraints.

If there is not existence of speed / altitude constraints in the trajectory, the PAT is computed using the equation 3.8. For this case, the distance is computed using the equation 3.3 and the speed is a constant parameter equivalent to the maximum aircraft cruise speed[6].

If the case is the trajectory involves a speed altitude constraint, it is necessary to compute the PAT by using the following equation[7]:

$$PAT = PAT_{CS} + PAT_{NCS} \tag{4.1}$$

Where PAT_{CS} is the predicted arrival time from the initial position until the point where the constraint altitude *(CAP)* is achieved, and PAT_{NCS} is the predicted arrival time from the point where the constraint altitude *(CAP)* is achieved and the target waypoint.

Hence, the trajectory of the aircraft is divided into two sections as shown in the Figure 4.6.

Initially, with the conditions of a normal flight controlled by the FMS-4D and the equations explained in the Chapter 3, it is not possible to know the exact components

---

[6] *The maximum cruise speed is a parameter defined in the FMS-4D and depends of the aircraft model, and other technical specifications.*

[7] *This equation is used for any situation where the aircraft trajectory involves speed / altitude constraints. Despite of the Figure 4.6 shows an ascending situation, this procedure is applied for descending situations as well.*

of the point *CAP* (latitude and longitude) or the distance from the *CAP* point and the *Target Point*.

In order to figure out this problem, it has been decided that since the aircraft airspeed is constant in most of the cases while flying before the *CAP* point, therefore the vertical speed could be *defined as a constant parameter* according to the aircraft specifications.

Hence, by knowing the vertical speed and the airspeed at which the aircraft would be flying from *Initial Point* to *CAP*, it is possible to predict the time when the aircraft crosses the constraint flight level and the distance at that point.

This means that the PAT_{CS} is calculated using a constant vertical speed as following:

$$PAT_{CS} = \frac{|Z_{IP} - Z_{CAP}|}{VS_{CS}} \cdot \frac{1}{\partial} \tag{4.2}$$

Where Z_{IP} is the altitude at the initial point, Z_{CAP} is the constraint altitude and VS_{CS} is the constant vertical speed defined by the aircraft specifications. Also ∂ is a conversion factor from minutes to seconds for the vertical speed.

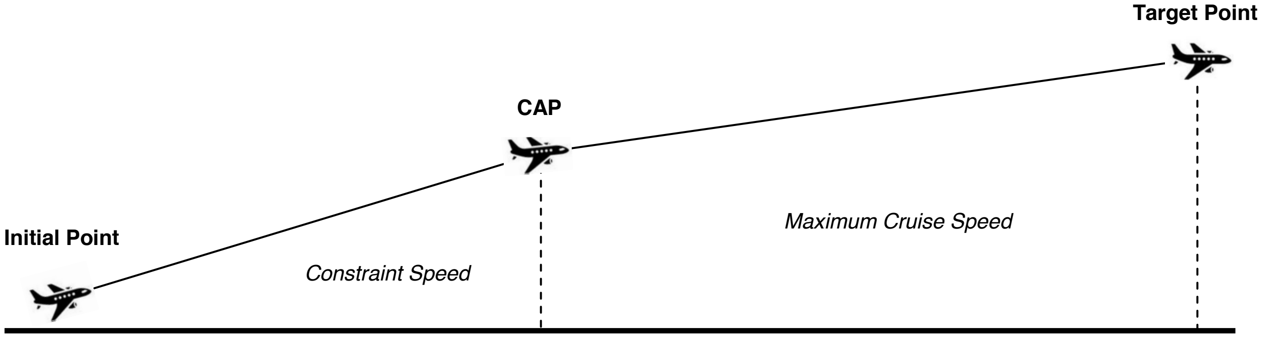

Figure 4.6: Trajectory of the aircraft used to compute PAT

Once the PAT_{CS} is computed, it is necessary to compute the other component of the equation 4.1.

Then, in order to calculate the value of PAT_{NCS} it is necessary computing the distance from *CAP* to the *Target Point* as follows:

$$D_{CAP\ to\ TP} = D_{total} - D_{IP\ to\ CAP} \tag{4.3}$$

Where D_{total} is a well-known parameter which is calculated using the Haversine equation 3.3 and $D_{IP\ to\ CAP}$ is calculated using the equation 3.8, isolating the distance $[x_{2-}x_1]$ and replacing the time with the PAT_{CS}.

Finally, the value of PAT_{CS} is calculated by using again the equation 3.8 using the *Maximum Cruise Speed* of the aircraft.

4.1.4.3. Weather Conditions

The weather conditions have been simplified in order to design the FMS-4D according to the information provided by the simulation environment.

According to X-Plane [15], there are three main layers defined by three different altitudes where the wind effects affect the aircraft in a different way (because they have different heading and speed).

The information provided by the simulation environment is being imported by the user and it is used by the FMS-4D according to the layer where the aircraft is involved. This means that the wind effect changes depending of the altitude of the aircraft and a procedure in the algorithm is constantly checking which layer should be used by observing the current aircraft altitude.

It is important to point out that this information is used as input of the wind effect model explained in the section 3.2.4.1 Wind Effect which is an important component of the equation 3.8.

4.1.4.4. Fuel Estimation

The fuel consumption of the aircraft is an important topic due to its relation with several problems for the environment, for the airlines and for their services.

In the FMS-4D design described in this thesis, it has been decided to estimate *–in a very raw way-* the Fuel consumption of the aircraft from parameters given of the aircraft specification. This information pretends to inform the crew about the quantity of fuel used to fly a specific path. However, this estimation does not pretend be exact and even less to go deep into the improvement of the aircraft performance and the fuel consumption.

It has been considered that this task has been performed in a very efficient way by the current FMS systems and these features can be inherited from the previous versions of these avionics systems. Instead of this, this project has been focused on decreasing the fuel consumption by performing navigation in four dimensions with the purpose of getting very low errors for the time of arrival, for the horizontal and vertical position. In addition, it is considered that these facts compose a real solution for the problem described in the motivation section of this thesis and contribute to avoid the waste of fuel of the aircraft while not performing holding patterns or any other procedures which affect their performance.

4.1.5. Program Structure and Flow

This section pretends to provide an overview of the program structure as well as describing important details related to the control operations performed by the designed FMS-4D.

Firstly, it is considered important to state that the program executes a set of functions. According to their purpose, the set of functions can be divided in the categories described in the Table 4.6.

Table 4.6: Functions classification

Functions	Description
Unit Conversion Functions	Since some input data is provided in different units than the requested for the output, it is used a set of functions to convert units.
Control Functions	These functions are used to computing control parameters in order to perform a correct 4D navigation of the aircraft. The control functions use the information provided by the flight plan (*Flight_Plan_Entry*) and the information received from the flight simulator (*UDP_Pack*), in order to create an output data (*UDP_Pack*). They are considered *the core* of the FMS-4D
Database Functions	These functions provide the communication interface between the *Flight_Plan_Entry* elements and the database. The relevant information concerning to the waypoint position as well as other information of the elements *Flight_Plan_Entry* is requested to the database
Display Functions	These functions are used to control the User Interface (UI) display. They are cosmetics elements that help to show the data to the crew but they do not interact directly with the navigation algorithms.
Pad Functions	These functions are used to manage the alphanumeric pads of the FMS. It is a cosmetic feature that does not interact with the navigation algorithms.
Miscellaneous Functions	These functions are used to change display settings, or secondary functions that support the performance of the main functions of the project.

The flow of the program is performed according to the loop shown in the Figure 4.7. In this figure is possible to point out the input parameters that are obtained from the aircraft sensors and in this case from the simulator through an *UDP_Pack.* Also it is possible to differentiate the output parameters that are sent to the autopilot panel in order to produce a change in the aircraft behavior.

The Control functions (explained in the section *4.1.3.2 Flight Control Parameters*) use the information provided by the input parameters in order to produce new output parameters.

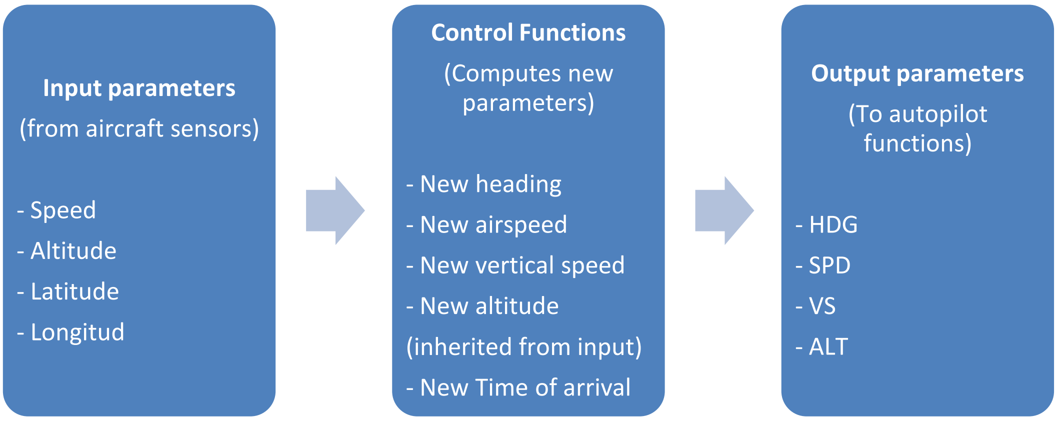

Figure 4.7: Program Flow

As mentioned above, these functions were already explained in the previous sections of this chapter. However, it is important to point out the relevance of the computation of the new predicted arrival time (PAT).

Since the control loop is a continuous procedure, it is important to update the current PAT by decreasing after each iteration a unit of time *equivalent to the timeframe chosen for the control system* (in practice, it is equivalent to 1 second). This means that every loop iteration the new PAT is computed as following:

$$PAT_n = PAT_{n-1} - STF \tag{4.4}$$

where STF is the system timeframe and n is the number of iteration.

Another important aspect of this section is related to the error of the system.

Every time the aircraft reaches a waypoint, it is important to count the difference of the current arrival time and the initial PAT for the reached waypoint, in order to add this error for the next path. In other way, the time errors of the previous waypoints would be added and produce a huge delay in the next paths.

Hence, if an aircraft flights from a Point A to a Point B and afterwards from Point B to Point C, the initial PAT of a trajectory from a point B to a point C is added the *remaining* or the *extra* time from the previous path as following:

$$PAT\_BC_1 = PAT\_BC_1 + PAT\_AB_n \tag{4.5}$$

where n is the last iteration of the previous path (A from B).

4.2. Hardware Design Overview

The FMS-4D hardware is basically composed by the same elements that compose a single FMS. These are a Control Display Unit (CDU) and the integration of the system with other avionics systems such as the Primary Flight Display (PFD), Navigation Display, and Multifunction Display (MFD) in order to perform tasks related to the control system, engine and fuel system, data link system and surveillance systems.

4.2.1. Processing information and memory aspects

The main function of the CPU is to perform the tasks related to the Flight Plan and Database access. Truly, this is a raw design concerning to hardware material, for this reason it is complex to select a specific processor. However, it is possible to determine possible features that should be taken into account for the processor selection.

Firstly, it has to be a low-consumption and high performance processor. The use of a power efficient processor is traduced to low aircraft consumption and this introduces the idea of using more than one processor for redundancy purposes with no risk of an excessively use of energy.

Concerning to the CPU, the processor executes critical tasks, for this reason it should be focused on executing several instructions per clock cycle. According to the processors used for the current FMS in the market, the CPU speed is no less 1.5 GHz and not consuming more than 10 watts of power[8].

The memory aspects of the current processors aim to values no bigger than 1 MB of L2 cache and the availability of using detection and correction of storage errors such as Error Correcting Codes (ECC).

4.2.2. Communication Bus Protocols and Software Interfaces

This section is aimed to explain a briefly idea regarding to the connection between the current FMS algorithms and de Application Software Interface ARINC 653.

As boarded in the previous section *4.1.2 Required Properties and Input / Output Data,* it has been designed a class for communication purposes which uses the UDP protocol to send and receive information to the aircraft. However, since the UDP format has some disadvantages (described in the section *3.3 Universal Datagram Protocol (UDP))*, it is important to have in mind that this communication interface is not the most efficient one for a real application.

---

[8] *This information has been taken from the PowerPC MPC7448 high performance processor technical specification sheet which is a typical processor used for Control Display Units (CDU) of Honeywell International Inc.[20]*

Because of this, the methods for sending and receiving data to the aircraft have been implemented into a single class. For this reason, it can be stated that this class could be replaced by another class for ARINC communication purposes in the future.

For these purposes, most of the commercial Flight Management Systems are provided with input / output plugs for the typical communication standards in aerospace applications such as MIL-STD-1553B, ARINC 429 and RS-232.

Chapter 5

TESTS AND RESULTS

5.1. Test Design

It has been designed and performed specific tests in order to know the general behavior of the FMS-4D algorithms. The testing phase has been divided in three sub-phases. Each phase has been performed several tests and evaluated relevant information according to the project advancement. The purposes of these tests are described as follows:

- Testing the general behavior of the FMS user interface.
- Testing the flight plan management features of the FMS (adding, removing, editing waypoints).
- Testing the time of arrival prediction algorithms (PAT estimation).
- Testing the speed, heading, altitude and vertical speed control systems based in the time of arrival prediction algorithms.
- Testing the general behavior of the FMS for the main phases of a commercial flight.
- Testing the exporting functions of the software developed.

In order to perform the tests, it has been designed a Flight with specific conditions, it has been exported the information of the flight and evaluated it using different tools and testing methods.

5.1.1. Flight Plan and Route Selection

In order to perform a test flight correctly, it has been designed a Flight Plan following the standard navigation procedures. The flight plan has been chosen taking into account the following statements:

- Short-range route but composed by ascending, cruise and descending phases.
- High-traffic commercial route
- Standard airliner cruise altitude and speed (according to a chosen aircraft)
- Including at least 6 waypoints with separation between 5 and 50 nautical miles.
- Including at least one of the following components: Fixes, Airports, and VOR or NDB.

Taking into account the requirements proposed above, the route chosen was from Frankfurt Main Airport (EDDF) to Amsterdam Schiphol Airport (EHAM).

Table 5.1: Flight Plan

Aircraft: B777-200	**Cruise True Airspeed:** 350 kts
Wind Effect: Yes	**Cruise Altitude:** 32000 ft

ID	FREQ	DIST	LATITUDE	LONGITUDE	NAME/REMARKS	TIME	ALT
EDDF	-	0	N50°01'59.89"	E008°34'13.64"	FRANKFURT MAIN	**-**	-
MARUN	-	47	N50°49'16.20"	E008°40'18.99"	MARUN	**550**	20000
ARPEG	-	18	N51°00'59.73"	E008°18'22.81"	ARPEG	**187**	28000
BADGO	-	5	N51°05'45.63"	E008°14'08.49"	BADGO	**56**	30000
ABILU	-	22	N51°24'36.48"	E007°57'15.22"	ABILU	**222**	32000
ADEMI	-	5	N51°28'57.52"	E007°53'18.89"	ADEMI	**52**	32000
HMM	115.65	23	N51°51'24.71"	E007°42'29.86"	HAMM	**241**	32000
REGBU	-	30	N52°04'25.07"	E006°58'14.37"	REGBU	**310**	12000
RELBI	-	6	N52°07'06.00"	E006°48'49.00"	RELBI	**65**	10000
RKN	116.80	2	N52°07'59.51"	E006°45'49.96"	REKKEN	**21**	9000
EHAM	-	74	N52°18'29.00"	E004°45'51.00"	AMSTERDAM/SCHIPHOL	**-**	-

The prediction arrival time has been calculated by the FMS-4D following the procedures explained in the previous chapter taking into account the standard cruise true airspeed of the selected aircraft (350 kts).

The automatic flight performed by the FMS-4D has been taken into account between the waypoints MARUN (begin of ascending phase) to seconds before REKKEN (begin of approach phase).

In addition, the operations for taking-off, approaching and landing have been performed manually by the test pilot.

In some cases, it has been necessary the use of flaps or airbrakes in order to meet the required airspeed and the flight status. These are actions commonly performed by the crew during specific phases of the flight (e.g. descending, the use of airbrakes to reduce speed).

5.1.2. Exporting and Data Processing

For the data exporting, it has been developed functions that create a *Comma Separated Values* (*.csv*) file with the information concerning to the aircraft status and position. The time step used is one second (1s) per row data and the parameters used as exporting data are described in the following table.

Table 5.2: Exporting data parameters

Parameter	Description
PAT	Refers to the Prediction Arrival Time (PAT) of each leg composed by two waypoints.
Latitude	Aircraft Latitude
Longitude	Aircraft Longitude
Distance to Waypoint	The distance from the current aircraft position to the target waypoint.
Altitude	Aircraft Altitude
IAS	Aircraft Indicated Airspeed
TAS	Aircraft True Airspeed
Vertical Speed	Aircraft vertical speed
Absolute Time	It refers to the Simulation time

Secondly, it has been coded a MATLAB script in order to perform relevant plotting functions (Appendix C) using the exported information described above.

The MATLAB code has been created in order to plot and calculate the information concerning to:

Table 5.3: Plot Information for the Testing Phase

Plots	Parameters involved
Horizontal Plane Navigation	Latitude, Longitude
Vertical Plane Navigation	Altitude, Absolute Time
Predicted Time of Arrival / Distance	Arrival Time, Distance to Waypoint
Speed Control Output	TAS, Absolute Time

Table 5.4: Parameter information for the testing phase

Parameters Computed	Description
Prediction Arrival Time Errors	Difference between the flight plan Expected Arrival Time and the Practical Arrival Time
Distance Errors	Minimum of the difference between the aircraft position at final PAT and the flight plan waypoint position

5.2. Validation and Results

5.2.1. Horizontal Plane Navigation (Heading control)

The Latitude [deg] vs. Longitude [deg] plot represents the horizontal navigation behavior of the aircraft. It can be pointed out the constant changes of the aircraft heading in order to achieve the flight plan waypoints in an efficient way.

In the test simulation, the aircraft maintained the heading of the aircraft with a very accurate precision. When a new target waypoint had been followed, the heading of the aircraft changes[9] instantly to the new target. It has been detected slowly changes in the output of the heading control system few seconds after turning and intercepting the predicted course.

The following figure shows the changes of heading performed by the aircraft in the whole simulation as well as a briefly overview of the path followed during the simulation.

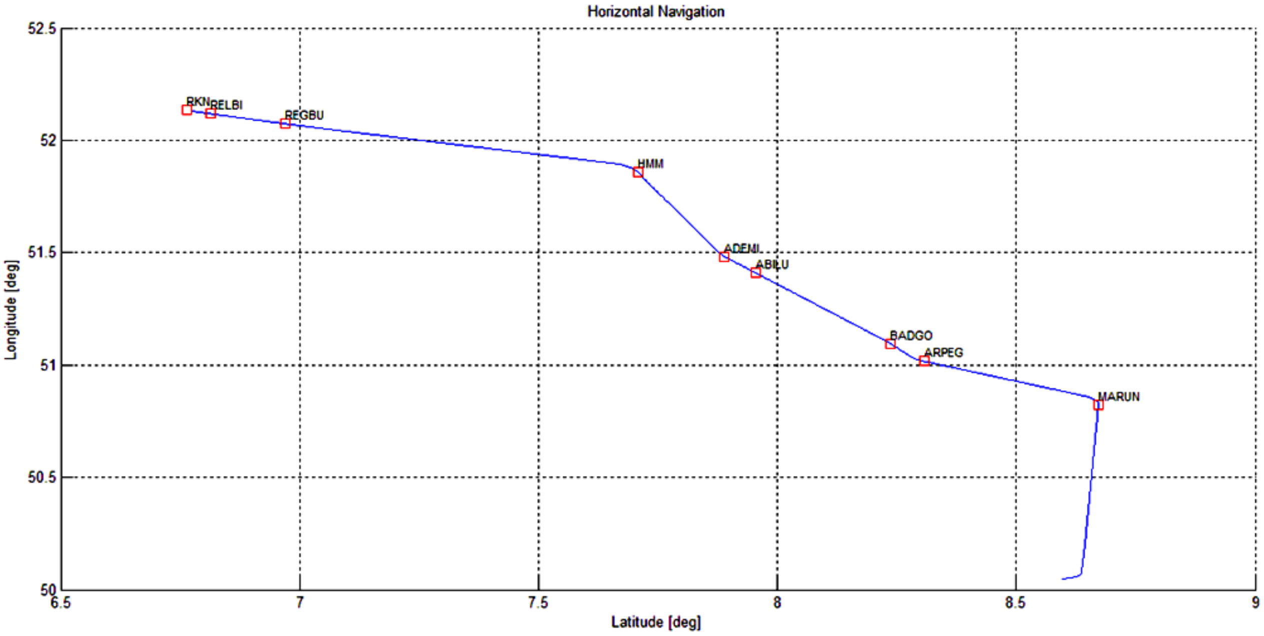

Figure 5.1: Horizontal Plane Navigation (Heading Control)

The Minimum Distance recorded to the Target Waypoint is shown in the Table 5.5. According to this information, the performance of the FMS in the horizontal plane has been achieved with a very low error in the route path. The maximum error is below 0.5 nautical miles and the minimum is around 0.3 nautical miles.

---

[9] *The bank angle of the aircraft has been set to 30 degrees showing a normal performance during the whole simulation.*

According to these results, it can be validated the efficiency of the heading control of the FMS-4D. The aircraft passed through all the Flight Plan waypoints with an acceptable very low error.

Table 5.5: Minimum Distance to Target Waypoint Results

Target Waypoint	Minimum Distance (nm)
MARUN	0,36793
ARPEG	0,32781
BADGO	0,34483
ABILU	0,49713
ADEMI	0,49901
HMM (HAMM)	0,38090
REGBU	0,39680
RELBI	0,41239
REKKEN (RKN)	-

5.2.2. Vertical Plane Navigation (Vertical Speed control)

The *Altitude [ft] vs. Absolute Time [sec]* plot represents the aircraft behavior for the vertical navigation. The FMS-4D controlled the aircraft in order to achieve the requested altitude in a smooth and regular way in the time required for all the waypoints.

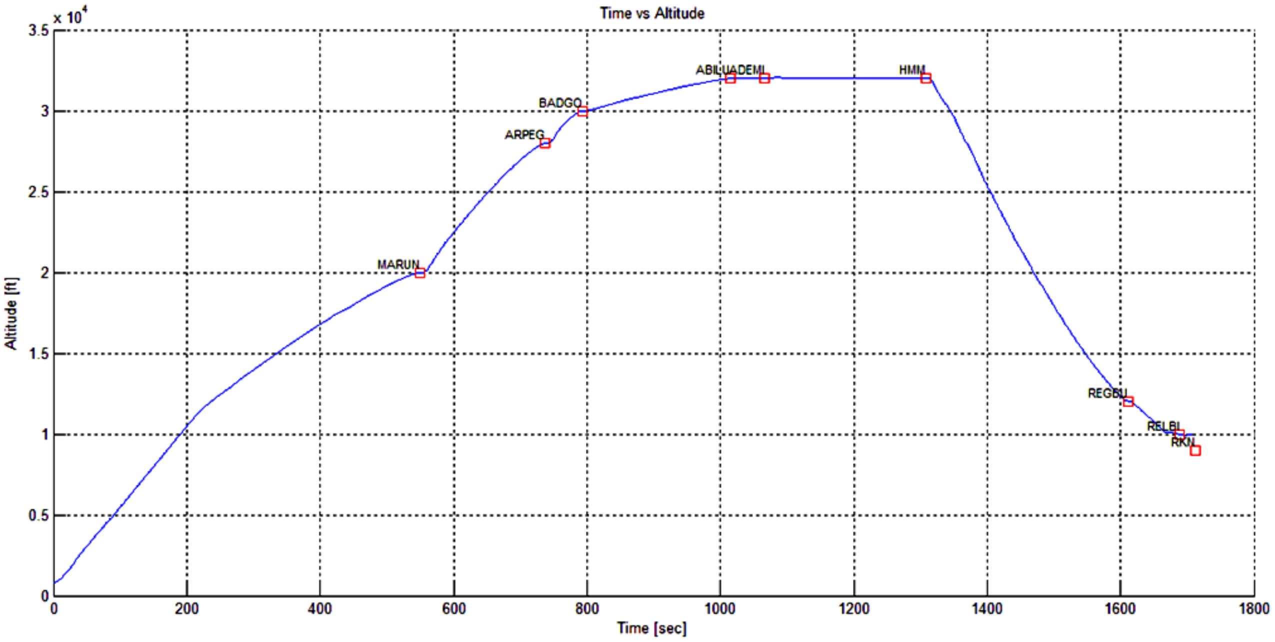

Figure 5.2: Vertical Plane Navigation (Vertical Speed Control)

Firstly, in the Figure 5.2, between the time 0 to the time at MARUN it can be clearly identify a two-steps ascending trajectory as studied in the section 4.1.4.2 *Minimum Arrival Time Prediction* and as described in the *Figure 4.6* of the same section. The point at 11.000 feet represents the CAP point. In the segment before this point the vertical speed and the speed remained constant at 3000 feet per min and in the segment after the CAP point till MARUN the vertical speed slightly changes according to the computation performed by the vertical speed control.

In addition, in the ascending phase it can be clearly defined three ascending phases with different climb rates. From begin of ascending point to MARUN (FL200), from MARUN to BADGO (FL30000) and from BADGO to ABILU/Cruise Level (FL320). It is clear the change of vertical speed in order to reach the desired altitude in the exact required time.

In the cruise phase the aircraft maintained FL320 with small changes of few feet due to the normal oscillation produced by the non-uniform air density or wind effects simulated by X-Plane and the altitude control adjustment performed automatically by the autopilot.

The following Table 5.6 shows the altitude values at minimum distance (shown in the previous Table 5.5) and its comparison with the Flight Plan altitude estimated for each point.

Table 5.6: Comparison between Practical and Flight Plan altitudes at target waypoint

Target Waypoint	Altitude at Waypoint (feet)	Flight Plan Altitude (feet)
MARUN	19981,77	20000
ARPEG	27951,78	28000
BADGO	29962,55	30000
ABILU	31976,63	32000
ADEMI	31984,88	32000
HMM (HAMM)	32000,08	32000
REGBU	12053,04	12000
RELBI	10003,30	10000
REKKEN (RKN)	-	-

According to the results shown above, it can be distinct a difference of few feet with respect to the flight plan altitude. This difference is considered a remarkably low error and it shows that the vertical speed control has been performed an outstanding labor in the vertical navigation plane.

5.2.3. Precision Time / Distance (Airspeed control)

The main aspect that differences a classical FMS to the FMS-4D is the feature of achieving a specific waypoint denoted by three-dimensional elements (lat, lon, alt) into a specific time of arrival.

As discussed in previous chapters, achieving a waypoint too early or too late (according to the required time) could produces the necessity of performing holding patterns in order to wait for another available time to arrive to the target waypoint.

For this reason, this FMS-4D has been designed focusing in providing airspeed control which allows the aircraft to reach the target waypoint at the required arrival time with high accuracy. In order to show the results of this objective, it has been decided to plot how the distance to the target waypoint decreases as the simulation time is closing to the precision arrival time.

In the Figure 5.3, it is shown how the relative distance to a target waypoint decreases with respect to the instant when the aircraft should arrive to the target waypoint (defined by the red squares).

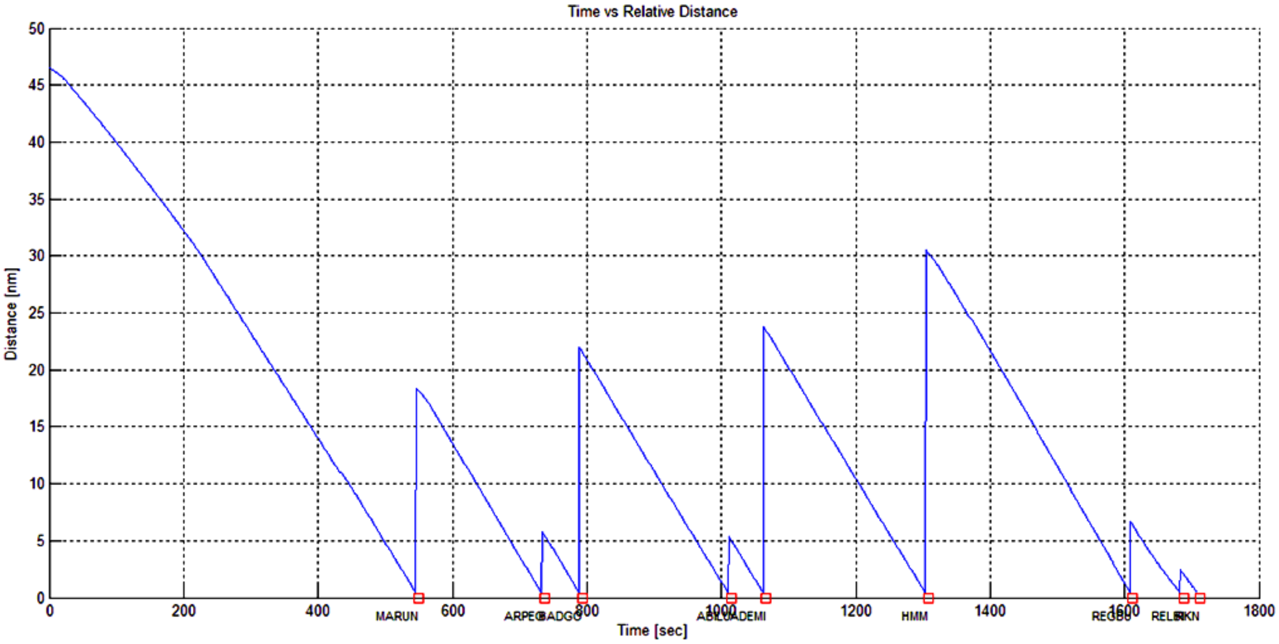

Figure 5.3: Precision Time / Distance Results

It can be observed that the relative distance decreases accordingly the simulation time is getting closer to the required time of arrival.

When the simulation point equals the required time, the distance to the target waypoint is almost zero. In order to show this information more precisely, the Table

5.7 shows how the arrival time matched accurately with the Predicted Arrival Time (PAT)[10].

Table 5.7: Arrival Time comparison with respect to the Predicted Arrival Time relative to the previous target waypoint.

Waypoint	Predicted Arrival Time	Obtained Arrival Time	Error
MARUN	550 (9:13)	548	2 (-)
ARPEG	737 (12:23)	735	2 (-)
BADGO	793 (13:16)	791	2 (-)
ABILU	1015 (17:14)	1012	3 (-)
ADEMI	1067 (18:11)	1064	3 (-)
HMM	1308 (21:11)	1306	2 (-)
REBGU	1612 (26:16)	1610	2 (-)
RELBI	1688 (28:02)	1685	3 (-)
RKN	1713 (29:14)	-	-

As shown in the previous table, the maximum error obtained was 3 seconds and the minimum one was 2 seconds. In all the waypoints, the aircraft arrives before than the required arrival time.

This result has been obtained by performing continuous control of the aircraft airspeed as shown in the Figure 5.4.

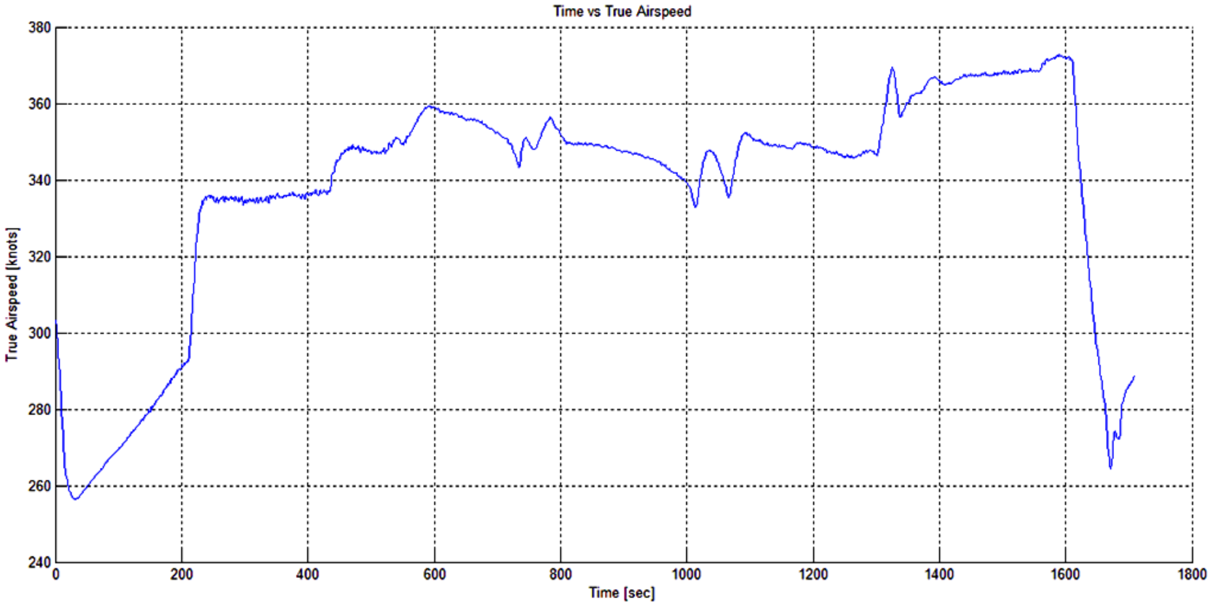

Figure 5.4: True Airspeed of the Aircraft

---

[10] *The Predicted Arrival Time and Arrival Time shown in the Table 5.5 are measured with respect to the simulation starting point (Simulation Time = 0).*

In the previous figure, it is possible to define three main segments. The first one is when the true airspeed increases due to the change of the aircraft altitude[11]. In this phase, it can be shown a fast change by the second 200. This is the CAP point, and the airspeed constraint was not taken anymore after this. For this reason, the airspeed increased quickly from 250 knots.

The second phase represents the segment where airspeed control was not constrained. It can be defined approximately from the second 200 till the second 1300.

As shown in the Figure 5.4, the true airspeed varies in a band of no more than ±15 knots out of the defined cruise airspeed (350 knots). The continuous change of airspeed is the main responsible factor that allows the aircraft matching the Prediction Arrival Time (PAT) with very low errors.

In the third phase, the airspeed increases for few seconds due to the start of descend and decreases quickly due to the same reasons of the first phase.

---

[11] *When the aircraft altitude increases, the true airspeed increases since this is calculated using the density differences at flight level and the sea level*

5.3. Issues and Errors

After several tests, numerous bugs, calculation errors and unit conversion problems have been detected and most of them have been resolved. This section describes the issues and errors detected in the system.

5.3.1. Predicted Arrival Time (PAT) for Short Distances

Several tests have been performed to the system. In the first testing phase, it has been detected some issues regarding to the computation of the Prediction Arrival Time (PAT) for small distances.

In these cases, the assigned PAT results to be too long for a short distance which induces the aircraft to slow abruptly the airspeed, even in some cases below than the stall speed V_S, producing a stall situation and disconnecting the autopilot system. The exact reason of this is associated to the small time the aircraft has to perform the airspeed control and prediction errors associated to the mathematical equations used.

In order to avoid this behavior is has been decreased the PAT for these points between 30% and 20% (values calculated empirically) and this PAT has been assumed as a discussed-with-ATC arrival time for the next testing phases.

5.3.2. User Interface (UI) and Database

The Control Display Unit (CDU) which represents the user interface of the designed system has a correct behavior. The input and output information is performed correctly with no-special details.

Since the input data type has not been fully validated, it is necessary to check the values types before typing them in the input fields. Probably, because of this issue it would be recommended to provide the user with previous training in order to use the FMS-4D.

The input information of a new waypoint is consulted in the database correctly with no special delays, the system fill-in the fields with the relevant information instantly once the user adds the waypoint ID into the relevant field, however, it has been detected that some existing fixes has not been found in the database. It would be necessary to update this database in order to solve this problem.

Fortunately, the information regarding to the chosen Flight Plan (FP) existed and have been used with no issues.

CONCLUSIONS

This chapter summarizes the thesis and shows an overview of the developed solution. In addition, it is described the difficulties faced during the project and the highlights of the results obtained. Finally, it gives an insight of the possible future work related to this project.

6.1. Overview of the System

In order to provide a possible solution to the problems faced by the aircraft in order to follow a four-dimensional trajectory, it has been designed a software-based Flight Managed System (FMS) that performs control of the flight parameters in order to achieve three dimensional waypoints in the required arrival time estimated.

The Flight Management System (FMS) is composed by display and miscellaneous functions used to control the user interface represented by the Control Display Unit (CDU) and it uses a database with simplified data regarding to real navigation data.

In addition, the system designed is composed by control functions that takes as input parameters that define the flight status and uses mathematical equations and estimation algorithms in order to calculate new values for heading, vertical speed and airspeed. By this way the aircraft is able to follow accurately a four dimensional trajectory defined by the Flight Plan (FP).

Also, the FMS-4D calculates a parameter defined as Prediction Arrival Time (PAT), which is an estimation of the time of arrival the aircraft takes to fly a specific path. The PAT is used to inform to the crew about the estimated performance of the aircraft in order to use this information to discuss the availability of the time constraints with air traffic management services.

In order to test the software it has been used a third-part flight simulator connected through a Universal Datagram Protocol (UDP) link.

The system has been provided with exporting data functions that output the information to be treated using MATLAB®.

In the hardware scope, it has been analyzed the designed system in order to suggest possible configurations about memory and processing.

6.2. Validation and Results Highlights

The FMS software designed shows that the defined User Interface (UI) that represents the Control Display Unit (CDU) provides clear and well-defined information about the actions performed by the system. Furthermore it provides important features to input data of the Flight Plan (FP).

The behavior of the database functions shows a correct access of the waypoints IDs requested by the user and the information provided about the input waypoints.

The system connects perfectly with the simulation environment using a UDP data-link and the information is exported correctly from the simulator.

The Flight Management System (FMS) control functions have been tested and the information has been analyzed in order to identify the performance of the aircraft in the horizontal plane (heading control), vertical plane (vertical speed control) and the time of arrival (airspeed control).

The test results show an overall robust system that controls the aircraft parameters in a quite efficient way. The trajectory described by the flight plan has been followed obtaining errors below 0.5 nm for the horizontal plane and below 3 seconds for the arrival time. In the vertical plane the altitude has been achieved in the exact expected times and maintained successfully. The altitude constraint has been respected correctly.

It has been identify some issues regarding to the validation of some input data. For this reason is advised the user is the actual responsible of the information provided to the FMS-4D.

6.3. Future Work

According to the results provided by the test performed, the Flight Management System (FMS) designed, controls the aircraft to follow a four-dimensional trajectory successfully. However, this software is very far from having all the features provided by a real FMS. Consequently, it has been identified several features that can be implemented to this project as a future work.

Firstly, the information of the Flight Management System has not been properly validated. Since this is a *first version*, it has been decided that the input data regarding to data type has to be checked by the user in order to make it consistently.

The output format of several parameters has not been standardized with the formats used in the aircraft systems. Position, Time and Heading values are often shown following a more mathematical schema instead of an aeronautical schema (The time values are shown in seconds instead of *mm:ss* [12] *format*, the position and the heading are shown in *signed degrees format* instead of in *DD MMM SS + compass direction format*[13]).

The flight constraints are an important feature to go in deep, in order to simplify things, the system provides to the pilot with the facility to add only one (1) altitude /

---

[12] *mm:ss refers to a format where the minutes and seconds are shown with two ciphers separated by ":" character. (e.g. 12:26 means twelve minutes and twenty six seconds).*

[13] *DD MMM SS + compass direction format uses Latitude from 0 to 90 and Longitude from 0 to 180. And the direction is assigned by using a 'N', 'S', 'E' or 'W' character. (e.g. 41°25'01"N and 120°58'57"W).*

airspeed constraint. It is suggested to improve this aspect, by adding more capacity to this feature or even creating new features such as the capacity for treating horizontal constraints.

It is suggested to improve the calculation of the Predicted Arrival Time (PAT) parameter in order to obtain more accurate values to allow the aircraft matching even lower time tolerances and to avoid the issues described with the low distances.

The aircraft navigates to a waypoint using a bearing computed directly to the target waypoint instead of following a specific leg-course or route. It is recommended to improve the flight control algorithm in order to carry out this feature.

The used communication protocol UDP has been used in order to create a link between the current system and the flight simulator. Conversely to the disadvantages of using UDP for critical operations, it has been used this protocol because it follows the standards of the third-part software X-Plane®. It would be interesting to implement a protocol that emulates the behavior of input / output data using an aeronautical communication bus protocol such as MIL-STD-1553B, ARINC 429 / 629 or even interacts with real-time interfaces such as ARINC 653.

BIBLIOGRAPHY

[1] EUROCONTROL, *“20-year Forecast of Annual Number of IFR Flights (2012 - 2035)*”. EUROCONTROL. (2013).

[2] “SESAR Factsheet: *I-4D - Flying a new dimension*” Single European Sky Research Team (SESAR). (2012). Nº 01/2012.

[3] “*Background on Single European Sky*” Single European Sky Research Team (SESAR). (2013). [Online] [Last Visited 26/01/14] http://www.sesarju.eu/about/background

[4] “*Air Traffic NextGen Briefing*” Federal Aviation Administration (FAA). (2009). [Online] [Last Visited 26/01/14] http://www.faa.gov/air_traffic/briefing/

[5] Gabriele Enea, Marco Porretta, “*A Comparison of 4d-Trajectory Operations Envisioned for Nextgen and Sesar, some Preliminary Findings*”. Engility Corporation. (2012).

[6] Karim Zeghal, Frank Dowling, “*4D trajectory management: an initial pilot perspective*”. EUROCONTROL. (2008). [Online] [Last Visited 26/01/14] http://www.eurocontrol.int/eec/public/standard_page/EEC_News_2008_2_TM.html

[7] Joel K. Klooster, Ana Del Amo, Patrick Manzi, “*Controlled Time-of-Arrival Flight Trials*”. Eighth USA/Europe Air Traffic Management Research and Development Seminar (2009).

[8] Carrie Sinclair, “*Honeywell Next Generation Flight Management System Meets New Air Traffic Requirements*”. Honeywell Aerospace Media Center (2010). [Online] [Last Visited 26/01/14] http://www51.honeywell.com/honeywell/news-events/press-releases-details/03.04.10HONNextGenNewAirTrafficRequirements.html

[9] R. Sabatini, S. Ramasamy, A. Gardi, T. Moore, C. Hill; L. Rodriguez Salazar, A. Adamou. “*Novel Four Dimensional Flight Management System with Enhanced Navigation Integrity Monitoring*”. Institute of Navigation GNSS+. (2013).

[10] Satish C. Mohleji, "*Optimal Flight Management System Utilization with ATC Automation*". IEEE AES Magazine, (1989).

[11] Virtual Skies. "*Air Traffic Management System: You Decide*"., National Aeronautics and Space Administration (NASA). [Online Image] [Last Visited 26/01/14]
http://virtualskies.arc.nasa.gov/atm/you_decide/arrival_rates.html

[12] R. W. Sinnott, "*Virtues of the Haversine*". Sky and Telescope, vol. 68, no. 2, 1984, p. 159.

[13] Ed. Williams, "*Aviation Formulary V1.33*" [Online Image] [Last Visited 26/01/14]
http://williams.best.vwh.net/avform.htm

[14] J. Postel, "*User Datagram Protocol*" Standards Documents. ISI. RFC 768. (1980).

[15] "*X-Plane Documentation*". Chapter 6. Section 6.1 "Data Input & Output". Laminar Research. [Online] [Last Visited 11/06/13]
http://wiki.x-plane.com/Chapter_5:_X-Plane_Menus#Data_Input_.26_Output

[16] Randy Walter, "*The Avionics Handbook: Flight Management System*". Smith Industries by CRC Press LLC (2001).

[17] "*FAA Federal Regulations. Part 91, Section 117*". [Doc. No. 18334, 54 FR 34292, Aug. 18, 1989, as amended by Amdt. 91-219, 55 FR 34708, Aug. 24, 1990; Amdt. 91-227, 56 FR 65657, Dec. 17, 1991; Amdt. 91-233, 58 FR 43554, Aug. 17, 1993]

[18] "*Our Airports: Data Downloads in CSV*". Megginson Technologies Ltd. (2013). [Online] [Last Visited 11/06/13]
http://www.ourairports.com/data/

[19] D. Richard Hipp, "*Appropiate Uses for SQLite*". Hwaci - Applied Software Research (2013). [Online] [Last Visited 11/06/13]
http://www.sqlite.org/whentouse.html

[20] "*Datasheet: High Performance MPC7448 PowerPC® Processor*". Freescale Semiconductor Inc. (2005)

A. ACRONYMS

3D 3-Dimensional
4D 4- Dimensional
4DT 4-Dimensional Trajectories
ACARS Aircraft Communications Addressing and Reporting System
ATC Air Traffic Control
ATM Air Traffic Management
CAP Constraint Altitude Point
CDU Control Display Unit
CO$_2$ Carbon Dioxide
CPU Central Processing Unit
CSV Comma Separated Values
CTA Controlled Arrival Time
DME Distance measuring equipment
ECC Error Correcting Codes
ETA Estimated Arrival Time
FANS Fuel Air Navigation System
FLXX0 Flight Level XX.000
FMS Flight Management System
FP Flight Plan
GE General Electrics
GNSS Global Navigation Satellite System
HP Holding Pattern
IAF Initial Approach Fix
IAP Instrument Approach Procedures
IAS Indicated Airspeed
ILS Instrument Landing System
LNAV Lateral Navigation
MFD Multifunction Display
NAS National Aerospace System
NDB Non-Directional beacon
NextGen Next Generation
NG-FMS Next Generation Flight Management System
nm Nautical Miles
PAT Predicted Arrival Time
PFD Primary Flight Display
RNP Required Navigation Performance

RTA Required Arrival Time
SAS Scandinavian Airlines
SESAR Single European Sky ATM Research
SID Standard Instrument Departure
STAR Standard Terminal Arrival
TAS True Airspeed
TOD Top of Descend
UDP User Datagram Protocol
UI User Interface
VNAV Vertical Navigation
VOR VHF Omnidirectional Range
WAAS-LPV Wide Area Augmentation System – Localizer Performance with Vertical

B. DEVELOPMENT SOFTWARE AND TOOLS

Adobe® Photoshop CS6 A graphics editing program used to create the User Interface (UI) images and others.

RouteFinder A online tool used to create flight plans.

MATLAB® R2012a a scientific software used to create plots and compute errors.

Microsoft® Office Excel 2007 A spreadsheet application used to manage CSV files.

Microsoft® Office Word 2007 A document application used to create this report.

Microsoft® Visual Studio 2010 A software development environment used for C# programming.

SQLite 3.0 A cross-platform C library that implements a self-contained, embeddable, zero-configuration SQL database engine.

SQLite Browser 2.0 b1 An editor for SQLite databases

StarUML 5.0.2 A software application used to create UML models and diagrams

X-Plane 9 A flight simulator used to test the algorithms and software designed

C. MATLAB CODE TO CREATE PLOTS AND ERRORS

```
% *****************************************************************************
%
% Project      : FMS-4D / Testing Phase
% Language     : MATLAB
% Initial Date : 31-10-2013
% Description  : Test.m MATLAB code that imports flight data from a xls and
%                creates plots and computes errors.
%
% Author       : Ing. Manuel Amaro (ma2c)
% Supervisor(s): Prof. Cristina Barrado (cb), Prof. Darius Rudinskas (dr)
% Write-access : ma2c, cb, dr
%
% --------------------------------------------------------------------------
%
% History
% -------
% 2013:
%   ma2c  Oct 31: introduced
%   ma2c  Sep 05: added general functions and flight data
%   ma2c  Nov 28: added errors computation
%   ma2c  Dec 10: added indicated airspeed plot
%
% 2014:
%   ma2c  Jan 17: updated with the most recent test data
%
% *****************************************************************************



% Flight Plan Waypoints

WAYPOINTS_NAME(1)=cellstr('MARUN');
WAYPOINTS_NAME(2)=cellstr('ARPEG');
WAYPOINTS_NAME(3)=cellstr('BADGO');
WAYPOINTS_NAME(4)=cellstr('ABILU');
WAYPOINTS_NAME(5)=cellstr('ADEMI');
WAYPOINTS_NAME(6)=cellstr('HMM');
WAYPOINTS_NAME(7)=cellstr('REGBU');
WAYPOINTS_NAME(8)=cellstr('RELBI');
WAYPOINTS_NAME(9)=cellstr('RKN');

% MARUN
WAYPOINTS(1,1)=50.821167;       % Latitude
WAYPOINTS(2,1)=8.672;           % Longitude
WAYPOINTS(3,1)=20000;           % Altitude
WAYPOINTS(4,1)=550;             % Time

% ARPEG
WAYPOINTS(1,2)=51.016667;
WAYPOINTS(2,2)=8.306389;
WAYPOINTS(3,2)=28000;
WAYPOINTS(4,2)=WAYPOINTS(4,1)+187;

% BADGO
WAYPOINTS(1,3)=51.096111;
WAYPOINTS(2,3)=8.235556;
WAYPOINTS(3,3)=30000;
WAYPOINTS(4,3)=WAYPOINTS(4,2)+56;

% ABILU
WAYPOINTS(1,4)=51.41;
WAYPOINTS(2,4)=7.954167;
```

```
WAYPOINTS(3,4)=32000;
WAYPOINTS(4,4)=WAYPOINTS(4,3)+222;

% ADEMI
WAYPOINTS(1,5)=51.482778;
WAYPOINTS(2,5)=7.888611;
WAYPOINTS(3,5)=32000;
WAYPOINTS(4,5)=WAYPOINTS(4,4)+52;

% HMM
WAYPOINTS(1,6)=51.8568992614746;
WAYPOINTS(2,6)=7.70829010009766;
WAYPOINTS(3,6)=32000;
WAYPOINTS(4,6)=WAYPOINTS(4,5)+241;

% REGBU
WAYPOINTS(1,7)=52.073631;
WAYPOINTS(2,7)=6.970658;
WAYPOINTS(3,7)=12000;
WAYPOINTS(4,7)=WAYPOINTS(4,6)+304;

% RELBI
WAYPOINTS(1,8)=52.118333;
WAYPOINTS(2,8)=6.813611;
WAYPOINTS(3,8)=10000;
WAYPOINTS(4,8)=WAYPOINTS(4,7)+76;

%RKN
WAYPOINTS(1,9)=52.1332015991211;
WAYPOINTS(2,9)=6.76387977600098;
WAYPOINTS(3,9)=9000;
WAYPOINTS(4,9)=WAYPOINTS(4,8)+25;

% ----------------------------------------------------

% Reading Aircraft Data
A = xlsread('export.xls');

% --------------------------------------------
% Figure (TIME VS DISTANCE)
figure;

% Figure labels and options
title('Horizontal Navigation')
xlabel('Latitude [deg]');
ylabel('Longitude [deg]');
hold on;
grid on;

% Plotting aircraft data
geoshow(A(:,2),A(:,3));

% Plotting flight plan waypoints
plot(WAYPOINTS(2,:), WAYPOINTS(1,:), 'rs');

text(WAYPOINTS(2,:), WAYPOINTS(1,:), WAYPOINTS_NAME, 'VerticalAlignment','bottom',
'HorizontalAlignment','left','FontSize',8);

% --------------------------------------------
% Figure (TIME VS RELATIVE DISTANCE)
figure;

% Figure labels and options
title('Time vs Relative Distance');
xlabel('Time [sec]');
ylabel('Distance [nm]');
hold on;
grid on;
```

```
% Plotting aircraft data
plot(A(:,10),A(:,4), 'b');

% Plotting Flight Plan waypoints
plot(WAYPOINTS(4,:), 0, 'rs');

% Adding text
ALenght=size(WAYPOINTS);
text(WAYPOINTS(4,:), zeros(1,ALenght(2))-1, WAYPOINTS_NAME,
'VerticalAlignment','top', 'HorizontalAlignment','right','FontSize',6);

% Computing the error distances
minimum_distance_error = 0.5;          % Set error to 0.5 nm;

vector_size = size(A);
number_waypoints = size(WAYPOINTS);
v = 1;

while(v < number_waypoints(2))

WAYPOINTS(5,v)  = 20000;                 % Adding a big value
m = 1;

while (m < vector_size(1))

    % Defining values
    lat1 = WAYPOINTS(1, v);
    lon1 = WAYPOINTS(2, v);

    lat2 = A(m, 2);
    lon2 = A(m, 3);

    R = 6371;

    % Converting Units
    lat1_rad = lat1 * pi / 180;
    lon1_rad = lon1 * pi / 180;

    lat2_rad = lat2 * pi / 180;
    lon2_rad = lon2 * pi / 180;

    % Computing diferences
    delta_lon = lon2_rad - lon1_rad;
    delta_lat = lat2_rad - lat1_rad;

    % Computing distance
    a = (sin(delta_lat / 2))^2 + cos(lat1_rad) * cos(lat2_rad) * (sin(delta_lon /
2))^2;

    % Computing Angular distance in radians
    c = 2 * atan2(sqrt(a), sqrt(1-a));

    % Distance
    distance_waypoint = c * R;

    %distance_waypoint = sqrt((A(m, 3) - WAYPOINTS(2, v))^2 + (A(m, 2) -
WAYPOINTS(1, v))^2);

    if(distance_waypoint < WAYPOINTS(5,v))              % Is it the lowest value?

        WAYPOINTS(5,v)  = distance_waypoint;            % Computing distance error
        WAYPOINTS(6,v)  = A(m, 10) - WAYPOINTS(4, v);   % Computing time error
(negative values are before, positive are after)
        WAYPOINTS(7,v)  = A(m, 5);                      % Adding altitude at that
time
        WAYPOINTS(8,v)  = A(m, 10);                     % Adding PAT (remaining) at
that time
```

```
        if (WAYPOINTS(5,v) < minimum_distance_error)     % If the distance error is
lowest than minimun error then stop calculating
            break;
        end

    end

    m = m + 1;
end

v = v + 1;

end

% --------------------------------------------
% Figure (TIME VS ALTITUDE)
figure;

% Figure labels and options
title('Vertical Navigation');
xlabel('Time [sec]');
ylabel('Altitude [ft]');
title('Time vs Altitude');
grid on;
hold on;

% Plotting aircraft data
plot(A(:,10),A(:,5), 'b');

% Plotting flight plan points
plot(WAYPOINTS(4,:), WAYPOINTS(3,:), 'rs');

% Adding Text
text(WAYPOINTS(4,:), WAYPOINTS(3,:), WAYPOINTS_NAME, 'VerticalAlignment','bottom',
'HorizontalAlignment','right','FontSize',8);


% --------------------------------------------
% Figure (TIME VS SPEED)
figure;

title('Time vs True Airspeed');
xlabel('Time [sec]');
ylabel('True Airspeed [knots]');
hold on;
grid on;

plot(A(:,10),A(:,7), 'b');


% --------------------------------------------
% Figure (TIME VS INDICATED AIRSPEED)
figure;

title('Time vs Indicated Airspeed');
xlabel('Time [sec]');
ylabel('Indicated Airspeed [knots]');
hold on;
grid on;

plot(A(:,10),A(:,6), 'b');
```