TABLE OF CONTENTS

LIST OF TABLES

Page

LIST OF FIGURES

Page

LIST OF ABREVIATIONS

5RA	Five routes algorithm
ATC	Air traffic control
ATAG	Air transport action group
APU	Auxiliary power unit
CO_2	Carbon dioxide
CO	Carbon monoxide
CCD	Climb/Cruise/Descent
CI	Cost index
EIG	Emission inventory guidebook
EICO	Emissions index of carbon monoxide
EICH	Emissions index of hydrocarbon
$EINO_x$	Emissions index of nitrogen oxide
EWK	Newark
FAA	Federal aviation administration
FF	Fuel flow
FMS	Flight management system
GRIB2	General regularly-distributed information in binary form version 2
GARDN	Green aviation research & development network
GS	Ground speed
HC	Hydrocarbons
ILS	Instruments landing system
IATA	International air transport association

ICAO	International civil aviation organization
ISA	International standard atmosphere
KIAS	Knots indicated airspeed
LTO	Landing to takeoff
LARCASE	Laboratoire de recherche en commande active, avionique et aéroservoélasticité
LNAV	Lateral navigation
LHR	London Heathrow
LAX	Los Angeles
MCL	Maximum climbing thrust
MNP	Minneapolis
NextGen	Next generation air traffic management system
PTT	Part task trainer
PDB	Performance database
RoC	Rate of climb
RRJ	Russian Regional Jet
RTA	Required time of arrival
SESARS	Single European sky
TOGA	Takeoff go around
TOC	Top of climb
TOD	Top of descent
TAS	True ground speed
WPT	Waypoint

US	United States
UTC	Coordinated universal time
VNAV	Vertical navigation
WA	Wind angle
WPG	Winnipeg
WS	Wind speed

INTRODUCTION

Lately, there has been a lot of concern in the aerospace industry about fuel needs and the polluting emissions generated by fuel consumption. There is a trend followed by many companies and airlines to deliver products that reduce fuel consumption. Many opportunities in saving fuel, thus polluting emissions, have been identified in the planning aircraft route.

Airlines have ground teams that search and identify the best routes for a given flight. The avionics equipment in the cockpit that helps the pilot to plan and to maintain a route is the flight management system (FMS). The main tasks that a FMS performs according to Collins in [1] are flight guidance, control of the lateral and vertical aircraft paths, monitoring of the flight envelope, computing the optimal speed for every phase of the flight and providing automatic control of the engine thrust, etc. In this work when "optimal" is mentioned, it means the value of the parameters that gives as result the lowest cost of a given flight.

Many different factors such as weather, traffic, or an emergency can change the predefined route given by the ground team. In these events, the crew has to determine a new route using navigation charts or existing FMS algorithms. FMS algorithms compute the "optimal flight altitude" and the "optimal speed". However, these algorithms need to be improved to find better routes and important data such as weather conditions have to be improved in order to take advantage of favorable winds.

The work in this thesis proposes a new algorithm that finds the optimal vertical navigation (VNAV) route in terms of speed and altitude. A Lateral Navigation (LNAV) route taking advantage of wind patterns is also proposed. A new method to calculate the costs of a missed approach in terms of fuel, flight time and fuel related polluting emissions is introduced. The VNAV and LNAV optimal routes are found by interpolating parameters in the performance databases (PDB) of the aircraft. For the calculations performed in this work, not only the total fuel required to perform a given flight is measured, but also the flight time. The cost calculations are a compromise between fuel burned and time related operations costs.

Required time of arrival (RTA) is not considered as a constraint. For the weather, real data was downloaded from the website of Environment Canada, and then these data were converted into a Matlab file and finally used to calculate the wind effects in flight.

The algorithm proposed does not perform an exhaustive search of all the combinations available in the PDBs to find the "optimal" VNAV profile. The algorithm reduces the possible combinations by defining a "pre-optimal" cruise profile in terms of altitude and speed. The algorithm searches and evaluates different PDB combinations around the "pre-optimal" cruise profile in order to find the "optimal" profile. Reducing the number of cruise combinations will reduce calculation time comparing with the exhaustive search method.

The trajectories calculated using this algorithm were complete trajectories; this means that climb, cruise, and descent were calculated to decide which trajectory from an initial point at the altitude of 2,000 ft to a final point at an altitude of 2,000 ft was the "optimal". All data needed for a FMS to guide the airplane is given as the output of the algorithm: KIAS/MACH climb profile, Top of Climb (TOC), cruise speed and altitude, Top of Descent (TOD), MACH/KIAS descend profile and the geographical coordinates that compose the trajectory.

The calculations performed in this algorithm were done for the Lockheed L-1011, from which the laboratory LARCASE has a full aerodynamic model via Flight-Sim, PDB and a FMS Part Task Trainer (PTT), and for the Sukhoi Superjet 100 (RRJ) from which the laboratory also has its PDB and the FMS PTT. All the PDBs were provided by CMC Electronics – Esterline.

The RRJ is a new airplane that started its service in April 2011. These aircraft were designed for medium and long flights. This algorithm focuses on this type of flights, and no optimization is performed for short flights (less than 800 nm). Even if the algorithm was tested and implemented in these two aircraft, it can be implemented in any airplane that has an available PDB.

For the new FMSs, optimizations of all routes are searched, that include the missed approach routes. Missed approach (or go around) is a procedure that is implemented when the aircraft has to abort the landing procedure. There are not many documented methods in the literature to compute the missed approach cost.

The work presented in this thesis starts with a literature review to justify and to expose the latest development of this area. Chapter 2 explains the different phases of a typical flight and the costs related to these phases. Chapter 3 describes the models used in this work such as the airplane model or PDB, the atmosphere models, the earth model, etc. In Chapter 4, a description of the calculation performed and considerations made in every flight phase are exposed. Chapter 5 illustrates the optimization method in VNAV used by the algorithm. In Chapter 6 the couple VNAV and LNAV is shown. During Chapter 7, a new method to calculate the missed approach costs is proposed. This calculation may help researchers in the development of algorithms to find the best route when a missed approach procedure is performed. Finally, in Chapter 8, results are presented.

The work presented in this thesis is part of the projects sponsored by the Green Aviation Research & Development Network (GARDN). This project is in collaboration with Esterline - CMC electronics. The name of the project registered from CMC electronics to GARDN is “Optimized Descents and Cruise” in which the objective is to reduce fuel consumptions, thus to reduce emissions.

CHAPTER 1

LITERATURE REVIEW

1.1 Environmental and economical background

Since the first flight performed by the Wright brothers, the aerospace industry has been outstandingly developed. Starting from that unstable aircraft that flew 12 seconds, the aeronautical technology has improved and developed into impressive military aircraft able to cross continents without stopping such as the aircraft B2, or able to fly at speeds higher than sound speed such as the F-22. Research spacecraft have been built, and in some cases, they have even left the atmosphere, such is the case of the space shuttles and the International Space Station.

However, the military industry is not the only one taking advantage of these developments. Civil aviation has also well developed, from small size aircraft used to deliver mail in which many pilots' lives were lost, to bigger size and safer airplanes such as the A-380 and the 747. These aircraft allow people and cargo to travel between different destinations in a fast and effective way.

Because it is the fastest way to travel, air transportation is one of the preferred ways of traveling; the Air Transport Action Group (ATAG) in [2] estimated that in 2009 only in the United States (US), 704 million passengers were transported by air. This industry can see nothing but growth in the coming years. Boeing estimated that the growth from 2010 to 2030 will be of 5% annually around the globe. Calculations done by the the year 2030 suggested the existence of 5.9 billon passengers around the globe per year. But passengers are not the only ones transported by air; GARDN estimated that in the year 2010 the value of cargo transported by air was of US$5.3 trillion.

In order to meet the needs of such a high volume of passengers and freight, current airports will have to be upgraded and new ones would need to be constructed. Also more aircraft

would be introduced into service; IATA also suggested that by the year 2030 the number of aircraft in service will be of 45,000 around the world.

This high number of aircraft in service will result in a high need of fuel. With a volatile fuel cost, which trend is to be higher each year (In the year 2011, the average value for the Brent crude oil was of US$ 100, US$31 more than in 2010), the new technologies being developed in the aerospace industry target to reduce the fuel consumption in order to reduce the flight cost and improve profit.

In 2008, ATAG in [2] estimated that the most important airlines in the US consumed 19.7 billion gallons of fuel and the American Department of Defence consumed in addition 4.6 billion gallons of fuel to perform their required activities. By the year 2011, the fuel cost needed for the airlines was of 178 billion dollars; this cost is 26% of all the expenses of the airlines.

The needed fuel does not only mean less profit for the airlines, but most importantly: it means pollution. Among the principal emissions from the fuel burned are carbon dioxide (CO_2), the combination of nitrogen oxide and nitrogen dioxide (NOx) and hydrocarbons (HC). The CO_2 is one of the major greenhouse effect gases and its release to the atmosphere is pointed to be one of the principal causes of global warming. In the year 2011, 649 million tons of CO_2 where released to the atmosphere by the airplanes. Almost 80% of this CO_2 was released in flights longer than 1000 kilometers where there is no other practical way of traveling. Also 2% of all the CO_2 released to the atmosphere is attributable to aviation. HC also contributes to the greenhouse effect. As mentioned by Ravishankara et al in [3], NO_x is pointed to destroy the ozone layer. This dioxide is released at high altitudes, thus, it is more likely to reach the stratosphere where the ozone layer is located. Another emission that is worth mentioning is vapor water. According with Nojoumi et al in [4], vapor water at high altitudes can cause clouds and can act as a greenhouse gas.

The aviation industry is aware of the problem of emissions generated by fuel and proposed itself ambitious goals to reduce emissions in the upcoming years. IATA in [5] reported that since 1960, fuel consumption has been reduced in engines by 80%. However, the aviation industry aims to reduce the fuel consumption and the emissions generated by the aircraft. In 2008 the aerospace industry represented by groups such as IATA agreed to develop what they called the "four pillars" with the aim to reduce fuel consumption and emissions. The first pillar is "the operational process", such as the reduction of the auxiliary power unit (APU) usage and the weight reduction in flights. The second pillar is "the infrastructure". Airports are being built and upgraded to meet the new regulations proposed: the Next Generation Air Traffic Management system (NextGen) in the US and the Single European Sky (SESARS). The third pillar which has not yet been implemented consists in the "economic measures". The fourth and last pillar is "new technology", such as new materials, new aircraft designs, new engines, new avionic systems, etc.

Using the four pillars mentioned above, what the industry is trying to increase fuel efficiency by 1.5% yearly from 2010 to 2020. The most ambitious goal for the industry is the reduction of CO_2 to half its value from 2005 by the year 2050. Figure 1.1 shows the forecast of the CO_2 reduction and the influence of every pillar in the goal reduction. This figure also shows the effect of the CO_2 emissions if any of the pillars is not implemented.

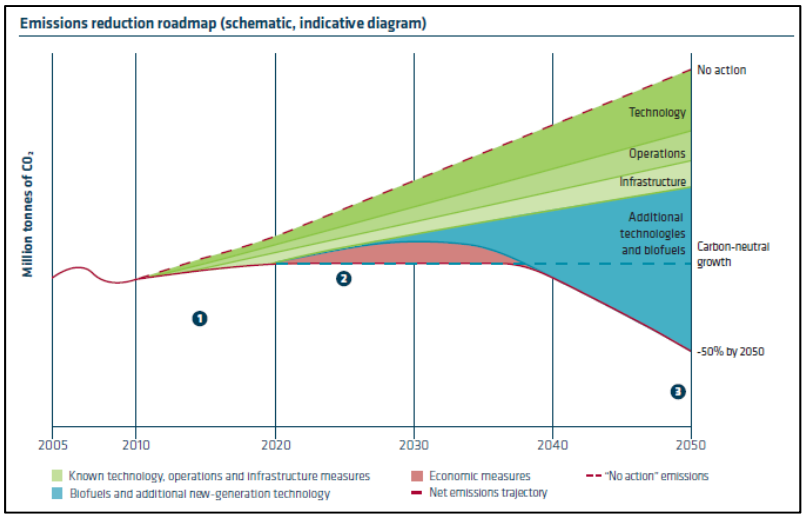

Figure 1.1 CO_2 emissions reduction roadmap
Source: ATAG Beginner's Guide to Aviation Efficiency (2010, p. 26)

Many associations such as the ATAG, the International Civil Aviation Organization (ICAO) and the Green Aviation Research & Development Network (GARDN) keep track and propose new technologies and methodologies to diminish the emissions.

Emissions and fuel are not the only parameters that these organizations are encouraging to reduce. The interest of noise contamination is also being taken into account. Even though, according to IATA in [5], the noise has been reduced by 75% from 110 dB in 1970 to 90 dB in 2010, it is desirable to reach the 80 dB which is equivalent to car noise at a street intersection or to a lower level.

1.2 Technological implementations

The aeronautical industry has already implemented some new technologies in the last years in order to achieve its ambitious goals. One of the most notable implementations is the winglets. Winglets are the folds at the tips of the wings. This change in the wing geometry helps reducing the magnitudes of the vortices generated by the difference of air between the upper and lower surface of the wing. Reduction of vortices will give induced drag reduction on an airplane. In some cases, winglets also increase the lift coefficient (C_L) on a given wing.

Boeing in [6] reported an economy of fuel up to 4.4% in a 3,000 nautical miles (nm) flight performed on aircrafts using winglets.

Airlines are also implementing programs to reduce fuel consumption and emissions generated. In the year 2006, as stated in [7], Air Transat added up improvements such as engines washing to improve their efficiency, changing the tires of the fleet to lighter ones, reducing the use of the auxiliary power unit (APU), taxiing with only one engine, reducing the weight of food related items, variation of the cost index (fuel to time ratio) during flight. The fuel consumption was reduced by 5% by Air Transat using the above implementations with some others.

Another technology being developed that would reduce the environmental problems is the biofuel. Different from fossil fuels, biofuels give a reduction of CO_2 in every single phase of their lifecycle. The plants that are ultimately used to generate the biofuel also absorb the CO_2 available in the air. Their utilisation, as shown in [8], has shown fewer emissions in comparison with the fossil counterpart. In Canada, as published by the Montreal Gazette in [9], with fundings from the GARDN program, Porter Airlines performed the first biofuel-powered passenger flight in 2012.

Different landing approaches have been developed to reduce fuel consumption such as the Continuous Descent Approach (CDA). The CDA is a type of descent in which the aircraft approaches the runaway in a continuous trajectory, instead to use a traditional step-down descent. IATA in [10] suggested an average reduction of 165 kg of fuel and 523 kg of CO_2 for a Boeing 767 in a single descent.

1.3 Trajectory optimization

There is interest in algorithm developments to obtain the optimal trajectory for a given flight. Linden at Honeywell was one of the first researchers that studied the trajectory optimisation for the FMS; his work was concerned mostly the 4D trajectory guidance, (guidance of the

aircraft to arrive at a destination with minimum fuel burn at a given time). In [11], the effects of tailwind, headwind and no wind in the flight cost were studied in the cruise regime by varying the speed, but no step climbs were performed. The optimal cost index was calculated by adding a penalisation if the RTA was not accomplished. This penalisation considered the costs of connection flights lost by the passengers. In [12], a study to determine the effects of the "step climb" in a cruise regime with and without meteorological conditions effects was done. Different methods were developed to determine when it would be the best time to perform a step climb. In [13], the effects of the "step climb" and winds were studied in the search of the optimal cost index for a long flight. A procedure was put in place to get rid of discontinuities in the time versus cost index relationship.

Hougton in [14] recommended parameters to identify such as cloud formation, temperature and usual locations of air currents to locate jet streams during aircraft flight. It discussed the benefits of flying with tailwind and the complexity of an airplane flight at its optimal altitude and the interception gain of a place among other airplanes in a jet stream flight.

Le Merrer in [15] applied the direct method of Herminte-Simpson collocation and the inverse dynamic programming to optimize the flight trajectory and then results obtained by both methods were compared. These methods were developed with the differential equations of an aircraft and optimal control concepts. The solutions obtained with both methods were found to be equivalent.

In our laboratory LARCASE, different optimisation methods for trajectories were found and published by Dancila et al in [16], Felix et al in [17], Gagné in [18] and Fays in [19].

Dancila et al in [16] proposed an algorithm using a PDB to find the best altitude in cruise; measured the time flight and the fuel flow for the A-310, L-1011 and the Sukhoi RRJ 100. In this algorithm, using the cruise trajectories were not divided in sub-trajectories as it is done in the FMS CMA-9000 of CMC Electronics - Esterline. The algorithm gives the same solution as the FMS of reference in 73% of the cases. In this algorithm only the cruise phase was

implemented for steady level flights. Climb and descent phases were assumed to have no effect on the flight optimal cruise altitude.

Félix et al in [17] proposed an algorithm using PDB tables found the optimal speed schedule and the optimal cruise altitude using the Golden Section methods for flight distances lower than 500 nm. For flight distances larger than 500 nm, this algorithm evaluated possible step climbs in ever waypoint defined in the route. In this algorithm, a combined mean optimization of 2.57% was attained for the A-310 and L-1011 aircraft with respect to the FMS CMA-9000 algorithm of CMC electronics. Nevertheless, this algorithm needed a complete analysis off all the available pair KIAS/MACH climbs and all the MACH/KIAS descents was needed, that made it time consuming. Besides, this algorithm did not consider the wind effects in the VNAV profile nor an evaluation of the LNAV.

Gagné in [18] proposed an algorithm that found the optimal vertical flight parameters by inspecting the complete trajectory. All possible combinations of climb, cruise and descent were analyzed to find the optimal one. In order to improve the cost reduction during cruise, the possibility of step climbs at each 25 nm was evaluated by measuring fuel flow. A precise method using weather forecast was developed to estimate in an accurate way temperature and wind effects in a flight. Nevertheless, it did not consider lateral navigation (LNAV). Besides, the high number of interpolations needed to perform all flight calculations and weather prediction made the calculation somewhat heavy. It is important to state that this work was a source of inspiration for this thesis.

Another work for the FMS trajectory optimization was developed by Fays. In [19], two algorithms were developed: one to avoid No-Flight-Zones (NFZ) and another one to find the optimal trajectory of an aircraft by combining the methods of *descent* and *tabou*. This algorithm was successfully implemented in a Boieng 747-400 and performed a trajectory from Montreal to Paris by avoiding obstacles placed at different altitudes, some in the aircraft trajectory and others outside the aircraft trajectory. The obstacles outside the trajectory were

chosen to prove that the algorithm would not suggest trajectories having obstacles on them.

1.4 Missed approach

In [20], an overview of aircraft trajectory management has been given that would produce noise reduction procedures. Noise produced by flying aircraft was modeled by using fuzzy logic as function of the received noise level during the trajectory, the sensibility of the areas being over flown and the time of the day when the aircraft departure took place. A nonlinear multi-objective optimal control problem was solved in order to find the best trajectory for a given scenario, aircraft and hour of the day. A practical example was given for the departure of an Airbus A340-600 from runway 02 of Girona International Airport. The methodology explained by Prats et al in [20] would assist airspace designers or airport authorities in order to implement noise reduction friendly procedures. The ATR aircraft are recognized in [21] as being the most efficient aircraft in their category, because of their high tech engines and propeller efficiency. The ATR 72-500 gives a 35% fuel saving per passenger with respect to an equivalent turboprop aircraft on a 300 nm average trip. In [21], the influence of flight operations on fuel conservation was examined, with the idea to give recommendations that will enhance the potential for fuel economy.

There have been studies on optimizing runways by maximizing the number of landings per hour in a given runway, in which the number of missed approaches was used as a tool to focus on minimizing costs such as the case of Jeddi in [22]. Nevertheless, a method has not determined in these studies to estimate the cost of missed approaches, instead, a constant value of $4,000 was selected to approximate that cost

CHAPTER 2

A TYPICAL FLIGHT AND ITS COSTS

The theoretical background to understand the ideas implemented in the new algorithm are described in this chapter. In Section 2.1, a typical flight is described so the reader can have a perspective of all the flight phases that have to be calculated in order to find the optimal profile. In Section 2.2, the cost of a flight and the concept of cost index are explained.

2.1 Typical flight

Every day, thousands of aircraft are crossing the sky around the world. All of these flights have different missions. The normal mission for a military aircraft is composed of many flight phases such as take-off, climb, cruise, supersonic dash, target approach at subsonic speed, needed turns around the target, cruise back, descent and landing. Commercial airplanes on the other hand have simpler mission trajectories. Commercial trajectories can be divided in three main flight phases: climb, cruise and descent and normally they do not come back to their departure coordinates as the military aircraft. Those three flight stages have sub-stages that will be explained in detail in the next sections of this chapter.

The quasi-steady flight phases described in flight obey to the equations of motion for an aircraft in translational motion. Because most of the times we are interested in steady unaccelerated flight, after some hypothesis discussed by Anderson [23], these equations can be expressed as

$$T = D \tag{2.1}$$

$$L = W \tag{2.2}$$

Where T is the thrust or power generated by the engines and it can also be seen as power, D is the drag of the aircraft, L is the lift generated by the wings and the airflow speed and finally W is the weight of the airplane. Figure 2.1 shows these forces acting on an airplane.

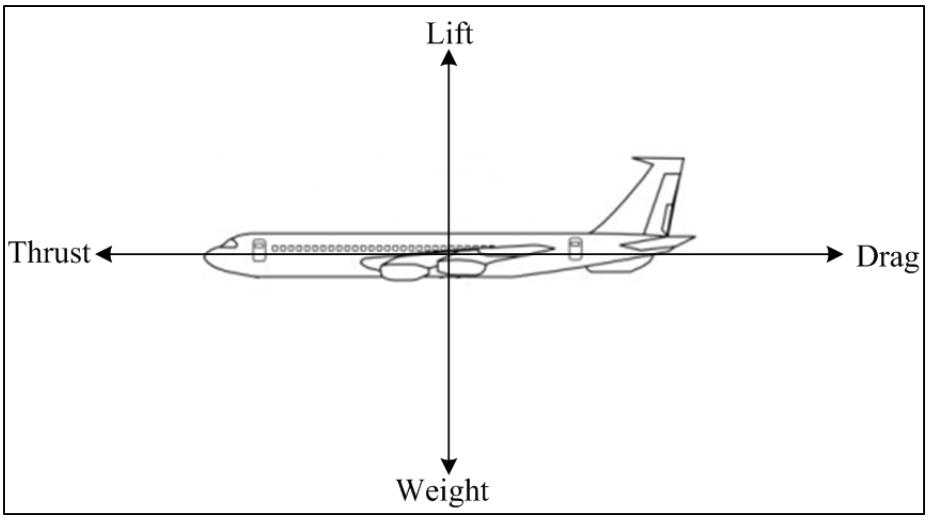

Figure 2.1 Steady flight forces diagram

Equation (2.1) means that to fly in a steady unaccelerated flight, the force generated by the engines (T) has to equal the drag forces (D) caused by the wind, the plane surfaces and the induced drag of the wings. If thrust happens to be less than the drag, a reduction of speed will be experimented; while if thrust is higher than drag, an augmentation of speed will be experimented by the aircraft.

Equation (2.2) implies that the force that pushes the airplane up (L) has to be equal to the weight (W) of the aircraft. If the lift is higher than the weight, then the airplane would begin to gain altitude. Otherwise, the aircraft would begin to lose altitude.

Equations (2.1) and (2.2) are highly coupled, thus a change in one of them will strongly affect the other one. A complete discussion of these equations can be found in the literature such as [23] and is not discussed in this work.

2.1.1 Climb

In a real flight, taxi and take-off are the first phases, but in this algorithm these phases are not considered because of the lack of experimental data and the regulations that change in many

airports around the world makes it difficult to create a generic algorithm. After take-off, the climb is the next phase, and it is calculated by the algorithm beginning at the altitude of 2,000 ft. There are many different engine climb configurations; the one used for the algorithm, is the Maximum Climbing Thrust (MCL). This configuration was chosen because it is the one that needs less fuel to climb than others. In this phase, the thrust generated by the engines is higher than the drag of the airplane because more power is needed in order to climb than it is needed to perform cruise. Also, this is the phase that requires the most fuel of all in a ratio of kg of fuel per nautical mile traveled. The reason is that the aircraft begins its flight at low altitude where the engines are less efficient, therefore more thrust is needed to find the solution of equation (2.1). Figure 2.2 is the force diagram for a typical climb.

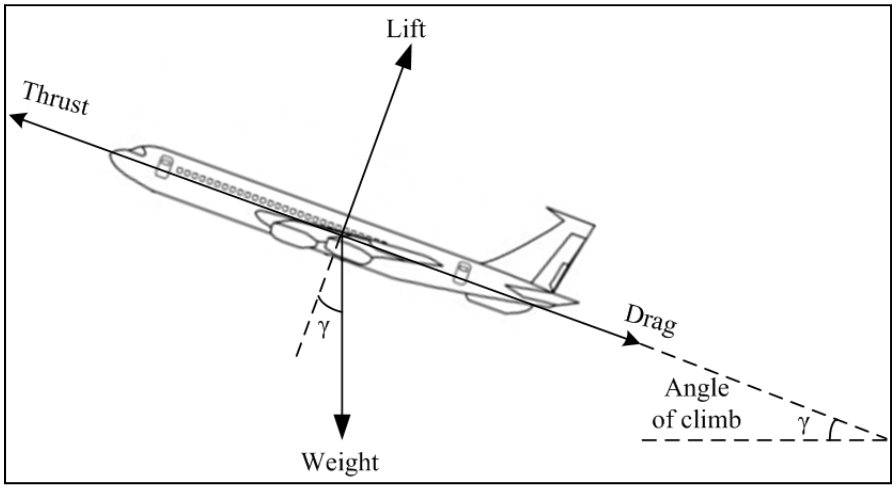

Figure 2.2 Force diagrams during a typical climb

By inspecting Figure 2.2, it can be seen that thrust does not only have to compensate the effects of drag, but also some of the forces generated by the weight. This means that more fuel will be needed to produce the needed thrust.

Equations (2.1) and (2.2) have to be changed to include the angle of climb (γ) effect; these equations take the next form:

$$T = D + W \sin\gamma \tag{2.3}$$

$$L = W \cos\gamma \tag{2.4}$$

Equations (2.3) and (2.4) express the influence of the weight due to the angle of climb of a given aircraft. The most interesting case is equation (2.3) because it is directly related to fuel consumption. If the angle of climb were to be 90 degrees, the weight and the drag forces would all be carried on and actually equal by the force generated by the engines (thrust) to gain more altitude. In a commercial flight however, this extreme situation will never happen. Still, the climb angle can reach levels of 20 degrees making this effect notorious by the fuel consumption.

In this phase the Rate of Climb (RoC) becomes evident. RoC is the vertical velocity of an aircraft and can be defined as:

$$RoC \equiv Aircraft\ Speed \cdot \sin\gamma \tag{2.5}$$

Equation (2.5) implies that the faster the aircraft flies at a given angle of climb, the faster it will reach the desired altitude, or TOC, the climb phase duration can be reduced to a minimum while increasing the aircraft speed. Nevertheless, climbing too fast may result in an expensive climb.

The climb phase is identified by its scheduled speeds, for example 280/0.78. The 280/0.78 means a constant climb at 280 Knots Indicated Air Speed (KIAS) and followed by a constant climb at 0.78 MACH (this speed change takes place after the crossover altitude) until the TOC is reached. Sometimes, in the beginning of a climb, the KIAS is lower than the one needed. In those cases acceleration will be performed to arrive at the desired KIAS before reaching the MACH climb. Figure 2.3 shows the typical phases of a climb that will be described in the next sub-sections.

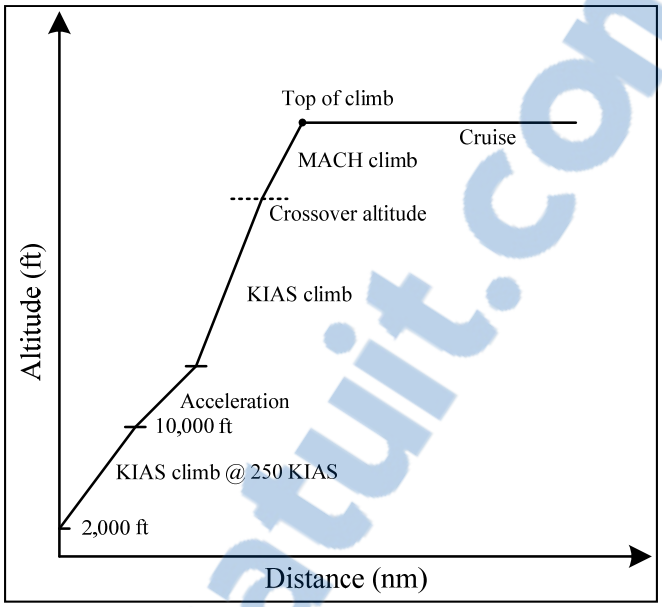

Figure 2.3 Different stages of climb

2.1.1.1 Constant KIAS climb from 2,000 ft to 10,000 ft

The climb phase begins after the take-off, and it is done at a constant KIAS. For the algorithm developed, this part begins at a given geographical point at 2,000 ft. At this altitude, according to [32] until the altitude of 10,000 ft, the aircraft cannot fly faster than 250 KIAS. For this reason the algorithm presented here will never exceed that speed, the aircraft speed will remain within those altitudes in this first stage.

2.1.1.2 Acceleration

When the airplane reaches 10,000 ft, the needed KIAS may be higher than the limit of 250 KIAS. Being that the case, more thrust will be needed to increase the KIAS of the aircraft to the KIAS needed.

2.1.1.3 Constant KIAS climb and the MACH crossover altitude

Once the targeted speed is attained after the acceleration, a constant climb is performed by the aircraft until the TOC is reached or until the MACH crossover altitude is reached, whichever happens first. When the MACH crossover altitude is reached, the crew may have to change the autopilot speed reference from KIAS to MACH. The MACH crossover altitude can be defined as the altitude where the true air speed (TAS) of KIAS equals the scheduled MACH number (in TAS) and depends on the scheduled KIAS/MACH profile climb. Mathematically, the TAS for a given MACH at a given altitude is expressed in knots as shown in equation (2.6) where c is the desired speed of sound in MACH and TAS_C is the speed of sound at a given altitude.

$$TAS\ for\ a\ given\ MACH\ number = c \cdot TAS_C(\text{altitude}) \tag{2.6}$$

It is really important to change the autopilot reference speed from KIAS to MACH, otherwise, once the MACH crossover altitude is surpassed, and more altitude is gained during the climb phase, the aircraft will fly faster than the expected KIAS, and the MACH would be closer to the speed of sound. Commercial aircraft are not normally designed to fly at such high speeds and fatal consequences may arrive if those speeds are reached.

2.1.2 Constant MACH climb

After the MACH crossover altitude, the climb continues at a constant MACH until the TOC or to the maximum altitude that the aircraft can reach. The speed of sound decreases with altitude that also varies with temperature. The speed of sound is proportional to the temperature which gradually descends with the altitude until the troposphere where it remains constant. The function that defines the speed of sound in a perfect gas is described in equation (2.8).

$$c = \sqrt{\gamma \cdot R \cdot T} \tag{2.8}$$

where γ is the adiabatic coefficient index of the air with an adimensional value of 1.4. R is the gas air constant with a typical value of 287 J/kg K, and T is the temperature of the air in Kelvin at a given altitude. It can be noticed that the speed of sound depends only on the temperature. The MACH number is calculated by dividing the actual TAS to the sound speed at a given altitude as expressed in equation (2.9).

$$MACH\ number = \frac{TAS}{c} \tag{2.9}$$

2.1.2.1 Cruise

The cruise is the most important phase of flight; it begins at the TOC and ends at the TOD. It is typically the longest part of the flight, where the most fuel is spent, and more opportunities of optimization exist. For every MACH, the aircraft has to provide the needed lift. During flight, the weight of the aircraft diminishes due to the fuel burned and affects equations (2.1) - (2.2), in order to keep their solution satisfied at a constant altitude-speed, the angle of attack of the aircraft has to be changed during flight. However, changing the angle of attack is a problem that it is not dealt here because the data available in the PDB considers this angle change.

There are three important things that strongly affect the fuel consumption during cruise: weight, speed and altitude. Equation (2.10), (2.11) and (2.12) show the relationship of the thrust with weight, lift and drag.

$$T = \frac{W}{L/D} = \frac{W}{C_L/C_D} \tag{2.10}$$

$$L = \frac{1}{2} \cdot C_L \cdot \rho \cdot V^2 \cdot S \tag{2.11}$$

$$D = \frac{1}{2} \cdot C_D \cdot \rho \cdot V^2 \cdot S \tag{2.12}$$

C_L is the aerodynamic lift coefficient which depends on the angle of attack. C_D is the aerodynamic coefficient of drag which is the sum of a fixed value due to the aerodynamics of the aircraft and the influence of the C_L (induced drag), ρ is the density of air, V is the speed of the aircraft and S is the area of the surface of the wing. Equation (2.10) directly relates weight with thrust. The ratio of lift and drag will normally be higher than one. Then it can be seen that the more the aircraft weighs, the more thrust will be needed, and thus more fuel. It is important to mention that when L/D is at its maximum the thrust would be at its minimum. Equation (2.12) shows that drag is directly proportional to the square of the speed, which means that the faster the aircraft flies, the more drag will be produced. Recalling equation 2.1 this affects directly the thrust needed, thus more fuel.

The last of the main factors that affect the airplane fuel consumption is the altitude. In equation (2.12), the density of the air is identified. The density of air diminishes at high altitudes, causing the drag to be lower at high altitudes, thus reducing the thrust needed. Figures (2.4) – (2.5) show the ideas explained above.

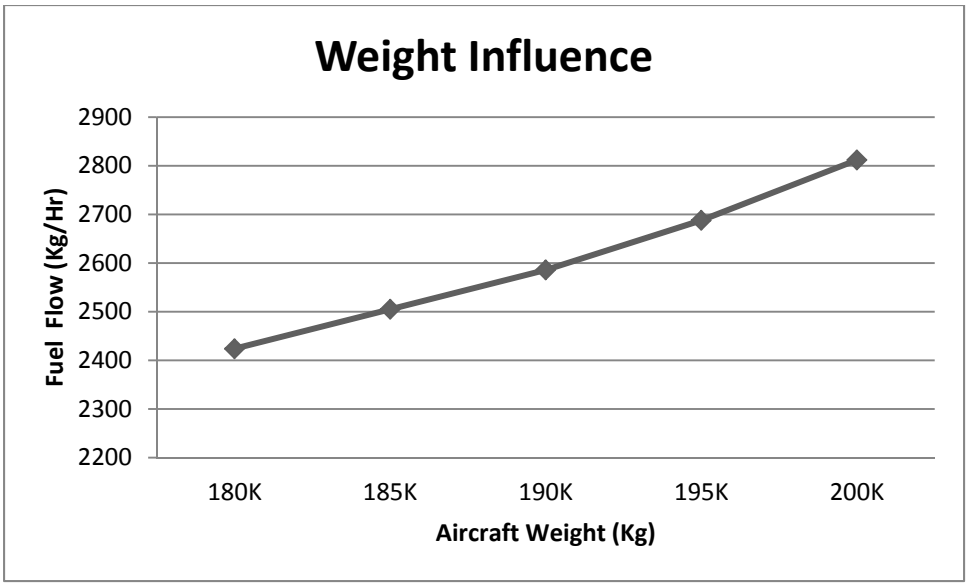

Figure 2.4 Weight influence on the fuel flow in a flight at constant speed and altitude

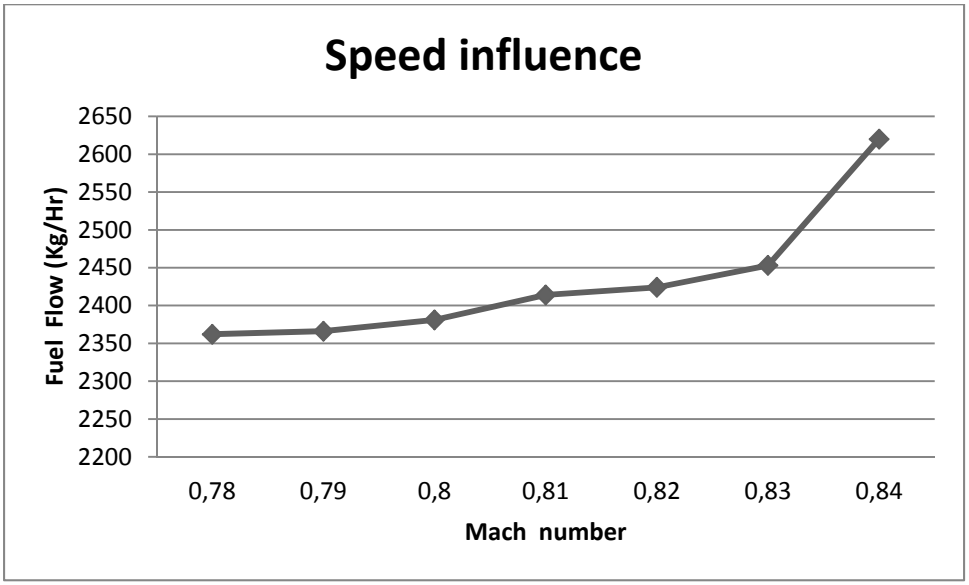

Figure 2.5 Speed influence on the fuel flow at constant altitude and weight

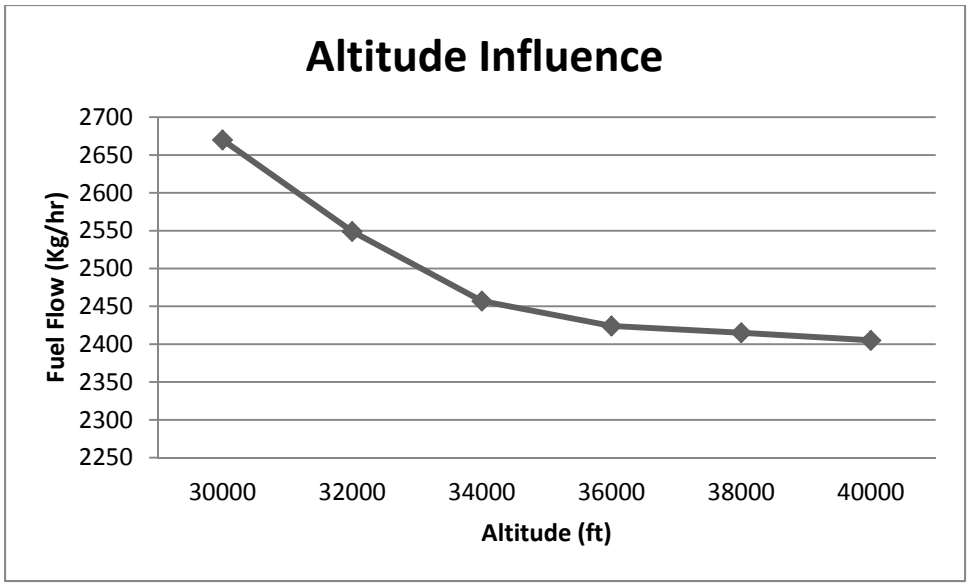

Figure 2.6 Altitude influence on the fuel flow at constant speed and weight

Figures (2.4) - (2.6) were traced with data obtained directly from the PDB of the L-1011. Figure (2.4) shows the influence of the weight on the fuel flow for a flight at 36,000 ft and 0.82 MACH. As expected, the fuel flow tends to be higher as weight is increased. Figure 2.5 shows the influence of the speed in a flight at 36,000 ft with a weight of 180,000 kg. As explained before, the faster the aircraft flies (MACH increases), the more fuel it needs to satisfy the equilibrium conditions at the desired speeds. Finally, Figure 2.6 shows the effect of the altitude on the fuel flow for a weight of 180,000 kg at 0.82 MACH. The higher the aircraft flies, the lower the fuel flow is.

As studied by Ojha [24], the ideal cruise is the one called climb-cruise. This cruise is not performed at constant altitude, but it climbs gradually as the weight of the aircraft is reduced. However, this kind of cruise cannot be implemented because it does not meet the current air traffic control (ATC) regulation, which requires the airplane to flight at a constant altitude and speed. Nonetheless, ATC may allow a climb to a different altitude after traveling a certain distance. This gives an opportunity to emulate the cruise-climb flight by changing altitudes during cruise. While flying at a given altitude, the aircraft asks authorization to the ATC to perform a climb to the next available altitude, and continues its trajectory at that new

altitude. This flight is called stepped-altitude flight and the climbs performed are called “step climbs”. These step climbs are normally performed for 2,000 ft or 4,000 ft climbs depending on the region, length of flight and the airline preferences. These climb steps are pairs in order to maintain the cruise in an even or pair altitude. In high traffic area even altitudes are assigned to traffic going to one direction and pair altitudes to the aircraft going to the other. Figure 2.7 is a graphical description of the constant altitude flight, climb-cruise flight and the stepped-altitude flight.

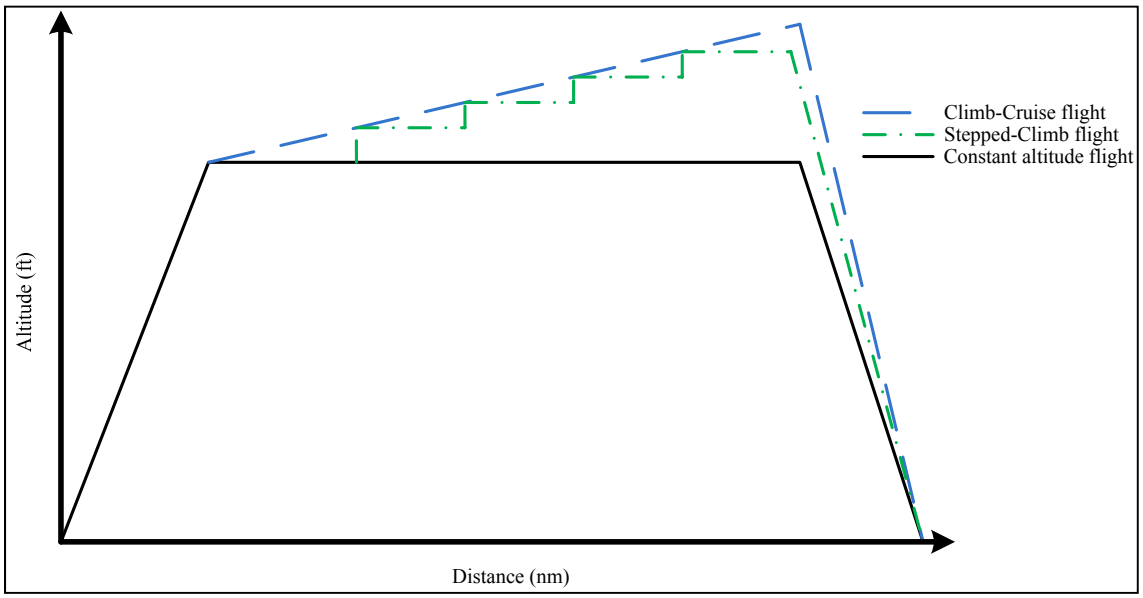

Figure 2.7 Comparison between different cruises.

2.1.3 Descent

This is the last phase of flight. It is also identified in a flight speed schedule as the climb phase, for example as MACH/KIAS/KIAS. Beginning the descent in MACH, the aircraft arrives at crossover altitude similar to that in the climb phase. After the crossover altitude, the crew has to change the speed to KIAS and then decelerate to a speed of at least 250KIAS at an altitude of 10,000 ft. It is the phase of flight in which the least fuel is spent. This is due to the fact that the lift of the aircraft diminishes allowing the aircraft to lose altitude. Because lift diminishes, the induced drag caused by the lift is reduced, and then the drag that has to be

generated by the engines is further reduced. Also the flight path angle (γ) makes the nose of the aircraft to descend below the horizontal as shown in Figure 2.8. This allows the weight to produce some a part of the forces to maintain the equilibrium in the system that in the other stages of flight would be produced entirely by the thrust. Figure 2.8 describes the force diagram for a descent flight where the thrust is present.

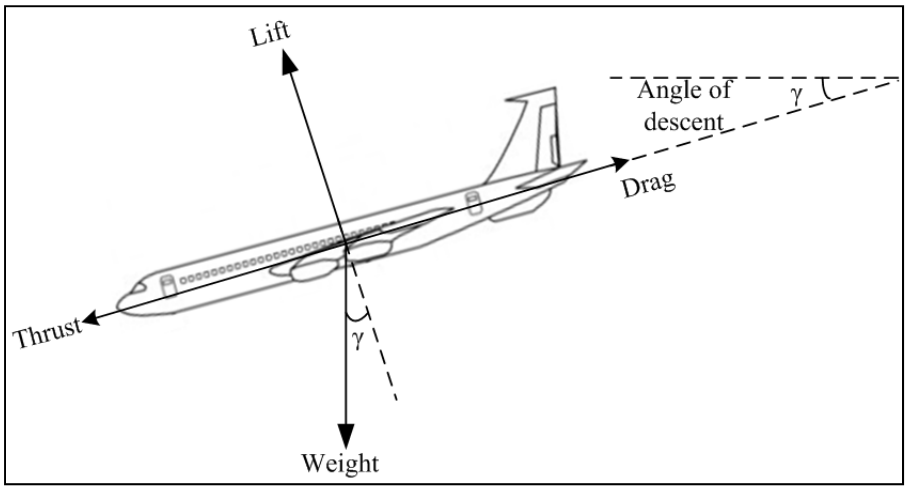

Figure 2.8 Force diagram during descent

$$T = D - W \cdot \sin\gamma \tag{2.15}$$

$$L = W \cdot \cos\gamma \tag{2.16}$$

Equation (2.15) shows the relationship of the weight with the thrust. In the case of engines failure, the aircraft can maintain the needed thrust (and reduce the lift losing rate) by selecting the proper γ angle. In equation (2.16), lift must be lower than weight because the airplane is descending, having an equal value would mean that the aircraft is maintaining the same lift as weight thus at constant altitude.

Basically, there are two different procedures for descent: the stepped-descent and the Continuous Descent Approach (CDA). During the first approach, the aircraft begins its descent to a given altitude and performs a small cruise, descending then to the next altitude, to maintain a short cruise and so on until the Instrument Landing System (ILS) altitude is

reached. This procedure is fuel consuming because it requires cycling the engines from idle to required thrust many times. In the second approach, the airplane descending angle is set approximately to 3 degrees, idles the engines and gliding descent to intercept the ILS altitude to finally reach the runway. The algorithm described in this thesis utilizes this last one to perform landing calculations. The CDA has been successfully implemented and tested in many airports such as Los Angeles (LAX), London Heathrow (LHR) and Newark (EWR). Figure 2.9 shows a graphic difference between these two landing approaches.

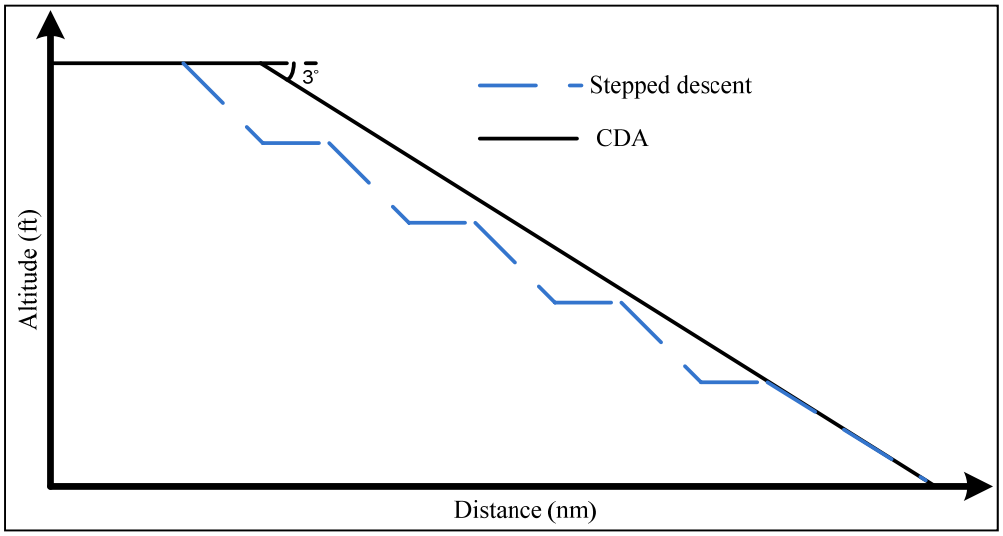

Figure 2.9 Comparison between the CDA and the stepped descent

Finally Figure 2.10 describes all the phases of the typical flight described in this work.

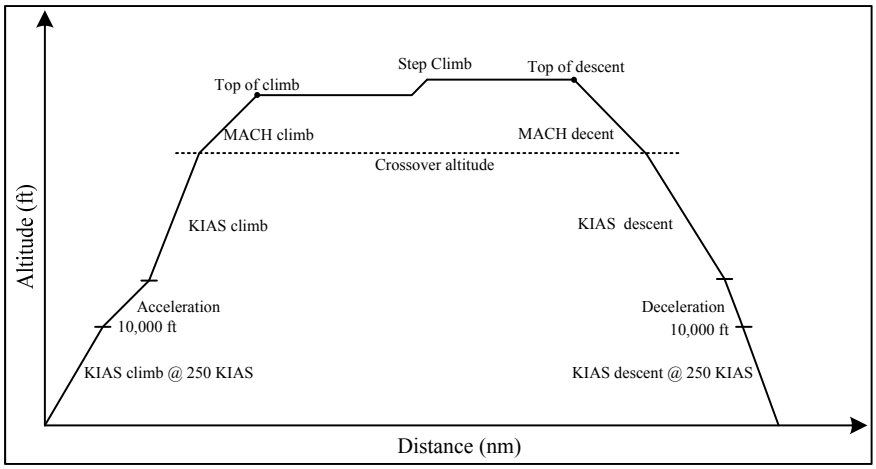

Figure 2.10 Flight phases

2.2 Total flight cost and the cost index

Flights cannot only be measured by how much fuel they need to fly on a given distance. There are many factors that influence the cost such as the salary of the crew, the maintenance cost of an aircraft, the cost of arriving too late or too early to a given gate, among others. A way to calculate the cost used often by airlines and by the FMS is the Cost Index (CI).

The CI allows a compromise between the cost of fuel and time related costs. A higher CI would give priority to a short flight time because the cost of time goes up, while a low CI would give priority to fuel consumption because the flight time is considered to be less important. The expression that defines the total flight cost is defined in eq (2.17) where the total cost, and the total fuel consumed is expressed in kg, the flight time (T) is expressed in hours and the CI in kg/hr, 60 is a conversion to minutes to be able to compare results.

$$Total\ Cost\ (Kg) = Total\ Fuel\ consumed\ + CI\ (kg/hr)\ \cdot T(hr) \cdot 60 \qquad (2.17)$$

While it is possible to change the CI in flight, in the work presented here it is always kept constant. In this work, when the word “cost” is used, it refers to the cost including the CI influence. The CI value is always selected by the airline and can change from one flight to another.

CHAPTER 3

STANDARD ATMOSPHERE, WEATHER AND AIRCRAFT MODEL

In this chapter, the International Standard Atmosphere (ISA) is described in terms of temperature, altitude and air density. The atmosphere model downloaded from Environment Canada, which is used to add the meteorological influence in the trajectory is described. Finally, the numerical aircraft model described by the PDB is explained and the way in which the interpolations are performed is shown at the end of this chapter.

3.1 The International Standard Atmosphere

The atmosphere is the mixture of gases that surround the earth. It is the transition between the land and outer space and it goes up to 100 km. Commercial flights are present from sea level up the troposphere (30,000 ft to 56,000 ft).

The most important values that are analyzed in the atmosphere for any given flight are: temperature, pressure and air density. Temperature is important because it affects the thrust of the engines; it also has a strong influence on the speed of sound and, in combination with the pressure, it fixes the value of air density. Temperature has a “zigzag” variation through the atmosphere. The atmosphere cools down from sea level until a given altitude, then it heats up, colds down again to finally heat up until outer space is reached. Pressure is used to determine the altitude and the speed of the aircraft, and as mentioned above, helps to fix the density of air. Density is the mass of air per unit volume and is dependent on temperature and pressure. It is one of the most important parameters in aircraft performance because it affects lift, thrust, and airspeed.

In order to have a standard platform to measure the performance of aircraft, ISA was created. The ISA models temperature, pressure, density and viscosity variation with altitude. It assumes there is no wind and clear weather conditions. In other words no rain, thunderstorms

or turbulence is considered. It is the ISA that is used in this thesis when no weather conditions are assumed. The model of the ISA described next is taken from [25].

Temperature, due to the zigzag behavior in the atmosphere is modeled in altitude ranges. Eq (3.1) describes the temperature from sea level to 36,000 ft, where T_0 is the temperature at sea level, which is 15 ºC or 288.15 ºK, T_h is the temperature lapse rate which is considered to be 6.5 ($ºF/1000\ m$) and h is the altitude in meters where the aircraft is located. After 36,000 ft, the temperature behaves somewhat constant and is considered to be -53.5 ºC or 219.5 $ºK$. Figure 3.1 shows the variation of temperature with altitude.

$$T = T_0 - T_h \cdot \frac{h}{1000} \tag{3.1}$$

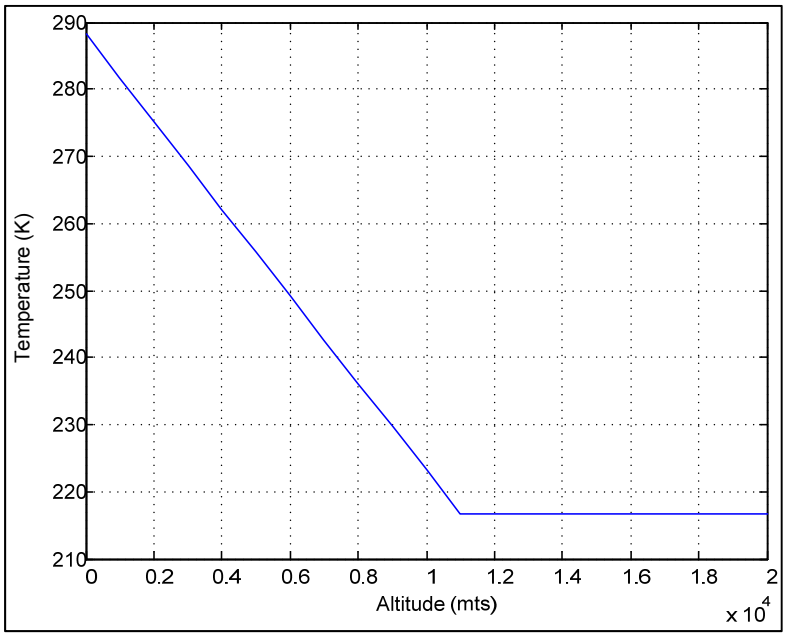

Figure 3.1 Temperature variation with altitude

In order to model the pressure in the ISA, the perfect gas law and the hydrostatic equation are manipulated to obtain the pressure at any given altitude. Pressure is expressed in equation (3.2), where g is the gravity acceleration of 9.8 m/s^2, T_h is the temperature lapse, P_1 is the pressure at sea level considered to be 101325 Pa, T is the temperature calculated by equation

(3.1), T_0 is the temperature at sea level and R is the gas constant of the air considered to be 287 J/ (kg) (°K).

$$P = P_0 \left(\frac{T}{T_0}\right)^{-\frac{g}{T_h \cdot R}} \tag{3.2}$$

Finally, the air density can be computed using the equation of state described in eq. (3.3) where ρ is the density of the air, P is the pressure at a given altitude, T is the temperature at a given altitude, and R is the gas constant of the air.

$$\rho = \frac{P}{R \cdot T} \tag{3.3}$$

3.2 Altitudes

There are many different altitudes, such as the geometric altitude, the absolute altitude, the pressure altitude, and the geopotential altitude. The geometric altitude is the altitude of an object above sea level. It is really important during climb, landing and approaching to high land such as mountains. The absolute altitude is the altitude from the center of the earth to the location of the object. The pressure altitude assumes a single pressure for every flight level. In this altitude, the crew must have the reference pressure provided by the ATC to locate the aircraft at a given altitude. Geopotential altitude is a transformation of the geographical altitude into an altitude that considers the reduction of the gravity caused by the increase altitude. It is mostly used for meteorological applications and it is described by equation (3.4), where h_G is the geopotential altitude, r is the radius of the earth typically with a value of 6357 km and h is the geometric altitude of the airplane.

$$h_G = \frac{r \cdot h}{(r + h)} \tag{3.4}$$

3.3 Airspeeds

There are many different speeds in aeronautics, such as TAS, KIAS, ground speed (GS) and MACH. This last one was defined in Section 2.1.2. KIAS is the speed that is measured directly from the speed sensor of the airplane (Pitot tube) and is directly read from the speedometer. TAS is the actual speed at which the airplane is actually flying within the atmosphere. The GS is the speed of the aircraft flying relatively to the ground.

TAS can be defined according to equation (3.5) where a_1 is the speed of sound at a given altitude in knots, γ is the specific heat of air, typically 1.4, P_0 is the stagnation pressure in the Pitot tube, and P_1 is the static pressure at a given altitude.

$$TAS = \sqrt{\frac{2a_1^2}{\gamma - 1}\left[\left(\frac{P_0}{P_1}\right)^{(\gamma-1)/\gamma} - 1\right]} \tag{3.5}$$

All the values of parameters found in equation (3.5) are available, except the stagnation pressure. Equation (3.6) describes this pressure where *Ps* is the pressure at sea level and IAS is the speed of the aircraft in knots.

$$P_0 = P_s\left[\left(\frac{IAS^2(\gamma - 1)}{2a_s^2}\right) + 1\right]^{\gamma/(\gamma-1)} + P_1 \tag{3.6}$$

The GS when airplane is flying in ISA conditions is the same as the TAS obtained in equation (3.5). However, if an atmosphere model that takes into account the influence of wind and temperature is used, the GS can be determined by adding the influence of the wind to the TAS. If the wind comes from the tail of the airplane, it makes the aircraft fly faster. On the other hand, if the wind is coming from the aircraft nose, it reduces the aircraft speed as shown in equation (3.7).

$$GS = TAS \pm Wind\ Speed \tag{3.7}$$

Normally though, the wind does not come directly from the tail or from the nose, but from different angles that change often during flight. In order to obtain the component of wind that pushes back or pulls forward the airplane, the wind vector has to be identified and decomposed in the longitudinal axis of the aircraft. This is not an easy task. In order to perform this decomposition, the wind triangle [26] is used which has been successfully implemented in [17][18][27]. Figure 3.2 shows the vectors involved in the wind triangle.

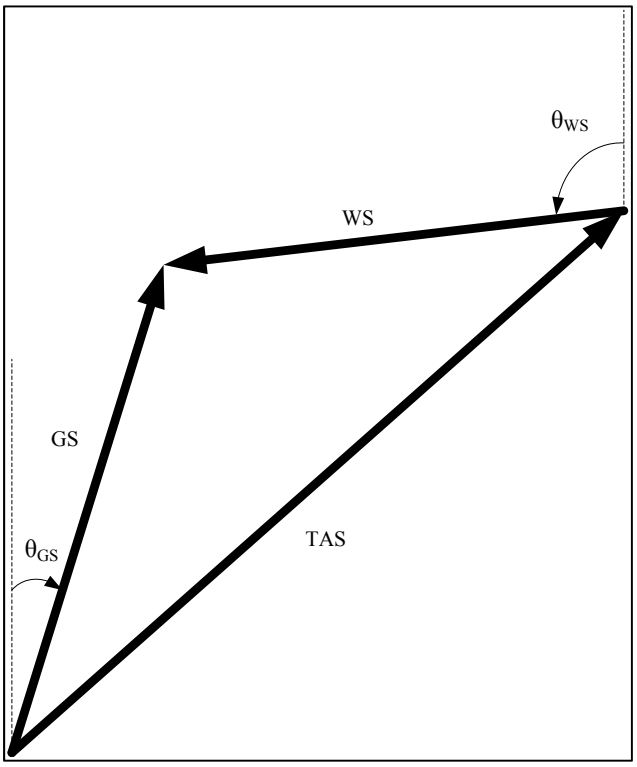

Figure 3.2 Wind triangle

By inspecting this figure, equations (3.3) – (3.5) can be written.

$$TAS^2 = TAS_x^2 + TAS_y^2 \tag{3.3}$$

$$TAS_x = GScos\theta_{GS} + WScos\theta_{WS} \tag{3.4}$$

$$TAS_y = GSsin\theta_{GS} + WScos\theta_{WS} \tag{3.5}$$

Where TAS_x is the component in *x* axis and TAS_y is the component in *y* axis from the vector *TAS*, *WS* is the wind speed, θ_{GS} is the angle to the destination point measured from the magnetic north (azimuth) and θ_{WS} is the direction of the wind measured from the magnetic north.

To obtain the GS, equations (3.4) and (3.5) are substituted in equation (3.3) obtaining a second degree equation (3.6) which can be easily solved using the general quadratic formula.

$$GS^2 - 2(GS)(WS)(cos\theta_{GS}cos\theta_{WS} + sin\theta_{GS}sin\theta_{WS}) + WS^2 - TAS^2 = 0 \qquad (3.6)$$

3.4 Earth Model

In the algorithm presented in this work, it is important to always know the position of the aircraft with respect to the Earth. This is important to correctly estimate meteorological conditions and to measure the distance traveled by the aircraft. The parameters needed from the earth model are the coordinate of longitude, the coordinate of latitude and the azimuth. The azimuth can be defined as the angle that is formed between the aircraft and Magnetic North. Even though there are many different models of the earth, the ones examined in this thesis were the function *legs, azimuth* and *track2* available with MATLAB and the equations of Vicenty implemented in two functions by Deakin in [28]. The MATLAB model and the equations of Vicenty provide the geodesic or great circle route. The geodesic route is the shortest curve between two points in a curved space such as the earth.

The function *legs* provides the distance between two points, the function *track2* provides the coordinates of a great circle between two points and the function *azimuth* gives the azimuth between two points. Vicenty's equations functions provide similar information as MATLAB functions. The two Vicenty'sfunctions are given by the direct and the inverse method. The direct method provides the coordinates where the aircraft is located after traveling a given distance in a given direction. The inverse method gives the distance between two points and the initial azimuth by providing the initial and last points.

The difference between these methods is the information that they need and the outputs that they can provide. Table 3.1 describes the Vicenty's equations methods.

Table 3.1 Vicenty's methods descriptions

Direct Method	
Input	Initial latitude (º), initial longitude (º), initial azimuth (º) and distance (meters)
Output	Final latitude (º), final longitude (º)
Inverse Method	
Input	Initial latitude (º), initial longitude (º), final latitude (º) and final longitude (º)
Output	Distance between points (meters), initial azimuth (º).

The formulation of these methods and the equations that describe them are explained by Gagné [18] and their complete development can be found in [28].

The Earth model selected for the algorithm is the one provided by the methods derived by Vicenty's equations. The reasons are that only 2 functions are needed instead of the three needed by the MATLAB model. Therefore, when the cost computation of the airplane trajectory is computed, the aircraft model gives the distance traveled to perform a task, e.g. horizontal distance traveled during a climb. The coordinates where the aircraft will be found are easily obtained using the direct method. The azimuth is found by using the inverse method. Only two functions are needed.

3.5 Weather model

The ISA is a good way of testing and developing algorithms and it is used to develop many different aeronautical technologies. However, for trajectory optimization it is not the most adequate model because real flights do not take place in conditions where the meteorological variables are standard and where winds are non-existent. Thus, a different meteorological model is needed to calculate trajectories for real flights. The current FMS from CMC Electronics - Esterline accepts up to 4 points in which meteorological data can be manually

introduced. This information is limited and it is not enough to search for alternative routes or to calculate the complete effect of weather in the flight cost. The obtainment of more meteorological information allows a better choice of a VNAV trajectory by searching the altitude with the best combination of temperature and wind. It also allows searching alternative lateral routes depending on the wind and temperature variations.

To obtain a precise model of the atmosphere, the global forecast from Environment Canada is used. This model provides meteorological information all around the Earth in the form of a grid. Every vortex that is shown in Figure 3.3 contains meteorological information [30]. This model is used because is precise, widely popular in North America, freely available and it has been successfully implemented in two other projects at LARCASE giving good results.

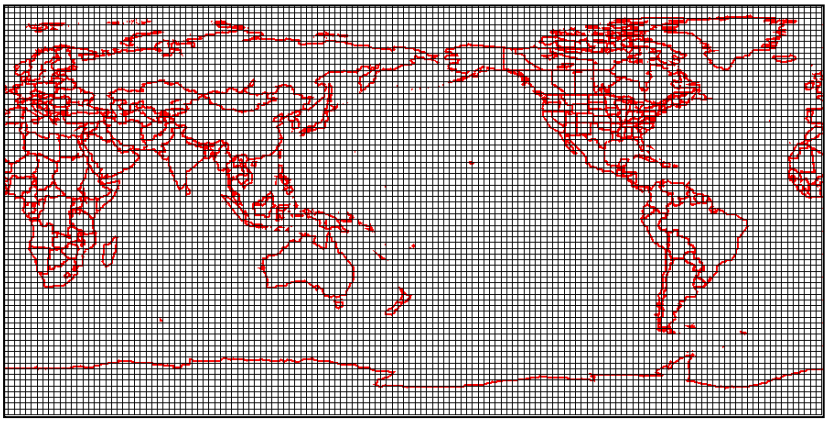

Figure 3.3 Global coverage of Environment Canada forecast
Source: Environment Canada

3.5.1 GRIB2 description

The information provided in this model is in the form of General Regularly-distributed Information in binary form version 2 (GRIB2). The GRIB2 files provide different information, but not all of it is needed by the algorithm. The needed information to perform our trajectory computation is available in the GRIB2 files that contain the data described in Table 3.2.

Table 3.2 GRIB2 variables needed in the algorithm

Variable	Variable Name	Units
TMP	Temperature	Kelvin
WDIR	Wind direction	Degrees
WIND	Wind speed	Knots
HGT	Geopotential altitude	Meters
MSL	Sea level pressure	Pascal

Firstly, this data has to be downloaded from Internet. Because many files have to be downloaded, a small script in Matlab was used to download the files automatically using the software wget [29]. The downloaded files have the following nomenclature [30]:

CMC_glb_Variable_LevelType_Level_projection_YYYYMMDDHH_Phhh.grib2

The meaning of each part of the file is described in Table 3.3. Time is in Coordinated Universal Time (UTC).

Table 3.3 GRIB2 file nomenclature

Chain Segment	Meaning
CMC	Canadian meteorological centre
_glb	GEM-GDPS Model
_Variable	Variable from Table 3.3 described in the file
_LevelType	Variable data at this Isobaric level
_Level	Isobaric level
_Projection	Projection used for the data. Latlon or polar. Latlon is the one used
_YYYYMMDD	Year, month and day of prediction
HH	Prediction time. Available every 3 hrs from hr 0 to gr 144
Phhh	P is a constant character and hhh the forecast hour
.grib2	File extension

There are many isobaric levels available in this mode. All quantities are in hPa: 1015, 1000, 985, 970, 950, 925, 900, 850, 800, 750, 700, 650, 600, 550, 500, 450, 400, 350, 300, 275, 250, 225, 200, 175, 150, 100, and 50.

For example a file named CMC_glb_TMP_ISBL_150_latlon.6x.6_20130201_P000.grib2 contains the temperature information from the GEM-GDPS global model at the isobaric level (geopotential altitude) of 150 hPA. The model in this file has the resolution of 0.6º x 0.6º. The forecasted date is February 1st, 2013. The forecast hour is 0h UTC.

Many GRIB2 files have to be downloaded in order to cover the complete route of the airplane with a meteorological forecast. Typically, the downloaded files are those that contain the variables described in Table 3.3 with the hours of prediction of the duration of flight for all the available isobaric levels. For example, if the aircraft takes off at 4:00 UTC, and the flight duration is 4 hours (flight expected to end at 8:00 UTC), the files downloaded in time are those that contain time frames of 3 UTC, 6 UTC and 9 UTC for all the isobaric levels.

3.5.2 Data conversion

The grib2 files cannot be read directly as they were downloaded, thus an interface is needed to be able to do this. One option is to execute a third party software to obtain the needed data of every point during the trajectory optimization calculation from the closest vortex in the grid, however, it is time consuming. Other option is to convert the information contained in the GRIB2 files to a .MAT Matlab file and have the information ready before performing the optimization calculations. This option is the one implemented in this work because it is faster and it allows having the meteorological data saved before the beginning of the optimization calculations.

The algorithm does not convert all the data in the GRIB2 files, since this conversion would take much time and not all the points of the grid are needed. To identify the data that has to

be extracted from the GRIB2 files, some possible points of the aircraft trajectory are generated. From all those possible points, the maximal and minimal longitude, and the maximal and minimal latitude are identified. With these minimal and maximal values, two points are created: one with the minimal values, and the other with the maximal values. These points represent the opposite vertexes of a box as seen in Figure 3.4.

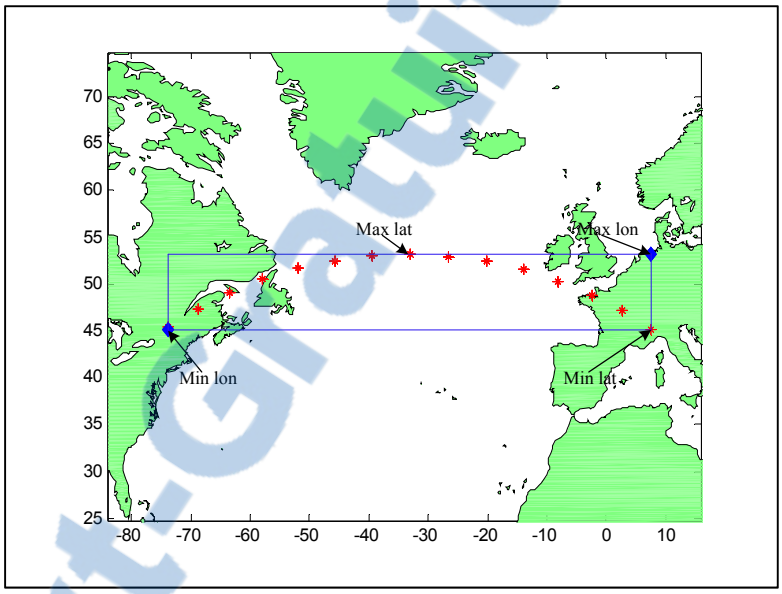

Figure 3.4 Maximum and minimal latitudes and longitudes in a given trajectory

Once the needed area of data is identified, the process of fetching the data starts. The MATLAB script takes advantage of a third party software called wgrib2 [31] which is used to fetch and decode the data from the GRIB2 files. This software has a function called –ijbox. This function conveniently converts and saves in a .txt file the data that is contained in a box formed by two points just as the ones described above. The .txt file created can be easily transformed to a .mat file in MATLAB.

3.5.3 Meteorological data interpolation

At this point, meteorological data is available for all the available geopotential altitudes, for the vertexes of a grid, and in three hour time blocks. In order to obtain the exact data to the

trajectory in a space/time frame, interpolations between the altitudes and the time are needed. Figure 3.5 describes an interpolation situation.

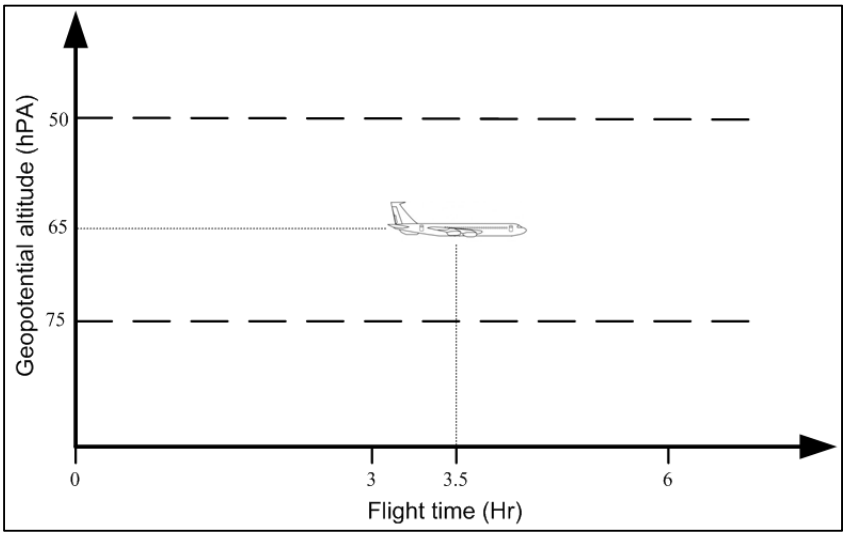

Figure 3.5 Interpolation situation for a given flight

The geopotential altitude 65 does not exist in the isobaric levels available, nor does the time 3.5 hr exist in the time blocks converted. In order to obtain those values, interpolations between the values that contain the required altitude (for ex. 65 hPa) and the required time (for ex. 3.5 hrs) are performed. For this work, from the four vertexes of the meteorological grid that surrounds the airplane, the point of the grid closer to the airplane location is the one selected to fetch the weather data from. This method was chosen among other methods such as the bilinear interpolation used by Gagné [18] and Gil [27] because it can reduce by 30% the calculation time. Also, the obtained estimations present minimal error as it is shown next in Table 3.4. Figure 3.6 shows the difference with the bilinear interpolations versus the closest point method.

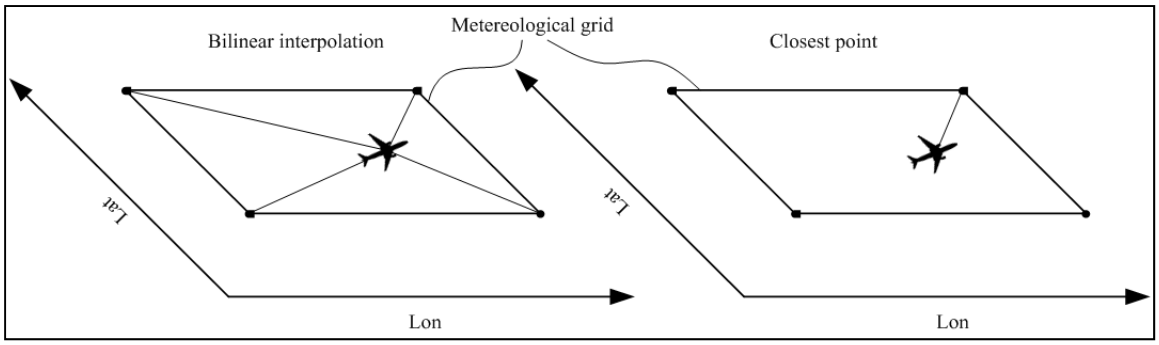

Figure 3.6 Difference between bilinear interpolations versus the closest grid point

Table 3.4 Comparison between closest point to grid versus bilinear interpolation

Temperature error	Wind speed error	Wind angle error
0.05%	1.53%	0.42%

With the altitude of the airplane located, the hour of flight calculated, and the closest point of the grid to the actual location of the airplane identified, the interpolation to obtain the weather information of the aircraft is ready to be performed. Figure 3.7 describes the path of the interpolation performed between two altitudes at a precise hour for a given variable.

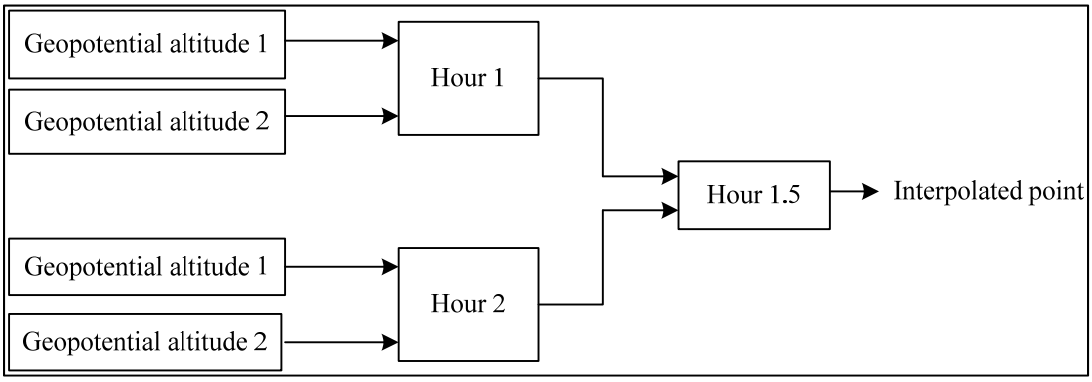

Figure 3.7 Weather interpolation path for a given variable

3.6 Aircraft model and the Performance database

In this section, the procedure to calculate and thus to obtain and calculate the fuel burned by the aircraft in a given trajectory is described. This procedure does not use the equations of motion of the aircraft. Instead, it uses a numerical model in the form of a Performance database (PDB) which is explained next.

3.6.1 The performance database

The information needed to perform all calculations is contained in the PDB. The PDB was provided by CMC Electronics – Esterline in the form of text files. These text files were converted to a MATLAB file (.mat) by the LARCASE student François Millet and then, the Sukhoi RRJ 100 PDB was adapted by the LARCASE student Adrien Charles Oyono Owono at LARCASE. The PDB can be divided in 7 sub-databases, one for every phase of flight described in Figure 2.10. In order to obtain the information from the PDBs, the input parameters have to be provided. Table 3.5 describes the inputs and outputs of the different sub-databases.

Table 3.5 Sub-databases from the PDB

Sub-database	Inputs	Outputs
Climb KIAS	KIAS (knots) Gross weight (kg) ISA deviation temperature (°C) Altitude (ft)	Fuel burn (kg) Horizontal traveled distance (nm)
Climb acceleration	Gross weight Initial KIAS (knots) Altitude when acceleration begins (ft) Delta speed to accelerate (knots)	Fuel burn (kg) Horizontal traveled distance (nm) Altitude needed (ft)

Table 3.5 Sub-databases from the PDB (continue)

Climb MACH	MACH Gross weight (kg) ISA deviation temperature (°C) Altitude (ft)	Fuel burn (kg) Horizontal traveled distance (nm)
Cruise MACH	MACH Gross weight (kg) ISA deviation temperature (°C) Altitude (ft)	Fuel flow (kg/hr)
Descent MACH	MACH Gross weight (kg) ISA deviation temperature (°C) Altitude (ft)	Fuel burn (kg) Horizontal traveled distance (nm)
Deceleration deceleration	Gross weight Initial KIAS (knots) Altitude when deceleration begins (ft) Delta speed to accelerate (knots)	Fuel burn (kg) Horizontal traveled distance (nm) Altitude needed (ft)
Descent KIAS	KIAS (knots) Gross weight (kg) ISA deviation temperature (°C) Altitude (ft)	Fuel burn (kg) Horizontal traveled distance (nm)

The databases have only certain parameters available for the aircraft flight envelope. For example, for Climb KIAS, the speeds are given in steps of 10 knots. The weight may be given by steps of 10,000 kg, the altitude by 1,000 ft steps, and the ISA deviation temperature might be given by 5 °C steps. These values cannot be introduced directly in the PDB. They need to be introduced by their indexes in the PDB. For example, for the KIAS, the index of 180 kts is 1; the index for 190 kts is 2 and so on. This happens for every single parameter. However, the output data is provided in the desired units, not in indexes. For example, in

order to find the fuel consumption and the horizontal distance traveled during a KIAS climb at 230 KIAS, with a weight of 205,000 kg, an ISA DEV of 0 ºC and at an altitude of 15,000 ft, the code without indexes would be written as follows (the numbers between the brackets are explained next): Climb_IAS.Output{2,1}(230, 205000, 0, 15000) = 1712 kg

The actual code to obtain the output of the aircraft with indexes is written as:

CLIMB_IAS.Output{2,1}(6,9,4,14) = 1712 kg
CLIMB_IAS.Output{2,2}(6,9,4,14) = 26 nm

If the input is needed, then the code is written as:

CLIMB_IAS.Input{2,1}(5) = 230 IAS
CLIMB_IAS.Input{2,2}(5) = 165000 Kg
CLIMB_IAS.Input{2,3}(5) = 5 ºC
CLIMB_IAS.Input{2,4}(5) = 6000 ft

Notice the number two after the output and the input command between brackets and before the coma ({*2*,1}). This number two (2) points the database where the data is found. If the number one (1) is introduced instead of two, some characters describing the column of the database are returned. Those returned characters are useless for the computations to be performed.

The second number between the brackets and after the coma points the database to what input or output information is being requested. In the output lines, it can be seen that when {2,1} is introduced, the output is in kg because fuel burned is requested and when {2,2} is introduced the output is in nautical miles because horizontal traveled distance is requested. The input indexes, {2,1}, {2,2}, {2,3}, and{2,4} access speed, weight, ISA deviation temperature and altitude data respectively.

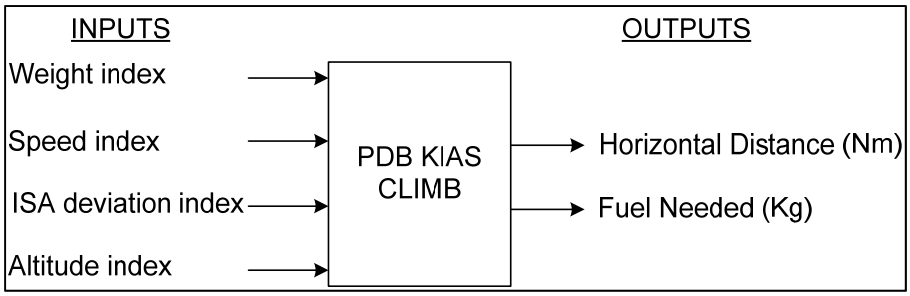

Figure 3.8 PDB output data fetching process

3.6.2 The performance database interpolation

Databases can hardly have all values that are needed. This is also the case for the PDB where the input information such as speed, weight, temperature and altitude are available by the value steps described in the previous section. In order to obtain data between those values, a linear interpolation has to be performed. The interpolation selected is the linear Lagrange interpolation, because it is simple and fast to implement. It is the same interpolation used by CMC Electronics – Esterline and it has also been successfully implemented in algorithms [17] and [18] developed at LARCASE and good results were obtained. Equation 3.7 describes the Langrage interpolation.

$$p_1(x) = \frac{x - x_1}{x_0 - x_1} f_0 + \frac{x - x_0}{x_1 - x_0} f_1 \tag{3.7}$$

In this work as requested by CMC Electronics - Esterline, interpolations between speeds and altitudes are not allowed. Weight and ISA temperature deviation are the only data in the PDB that can be interpolated. Nevertheless, during the acceleration and deceleration sub-databases it is allowed to interpolate in speed. This type of interpolation will be explained in the next chapter. Figure 3.9 shows how a typical PDB section in KIAS climb mode is looks like.

SPEED	100	IAS		SPEED	100	IAS	
WEIGHT	120000			WEIGHT 2	130000		
ISA_DEV	0			ISA_DEV	0		
	Altitude	Fuel burned	Horizontal distance traveled		Altitude	Fuel burned	Horizontal distance traveled
	2000	0	0		2000	0	0
	3000	68	0.6		3000	72	0.7
	4000	135	1.3		4000	144	1.4
	5000	**203**	**1.9**		**5000**	**216**	**2.2**
	6000	270	2.6		6000	287	3
ISA_DEV	5			ISA_DEV	5		
	2000	0	0		2000	0	0
	3000	72	1.2		3000	83	1.3
	4000	142	2.0		4000	150	2.2
	5000	**209**	**2.8**		**5000**	**220**	**3.1**
	6000	265	3.2		6000	291	3.4

Figure 3.9 Typical PDB data in mode CLIMB KIAS
Source: CMC Électronique - Esterline

Before performing the interpolations, it is important to determine the lowest and the highest steps within the database to obtain the needed values. A function was used in the code to identify the steps. For example if a value of temperature of 3.5 ºC is needed when the available values in the PDB input are -5 ºC, 0 ºC, 5 ºC and 10 ºC, the Matlab function will identify the lower step value to be 0 ºC and the higher step value to be 5 ºC. The interpolations will then be performed within these limits or steps. Figure 3.9 describes the normal interpolation path in the algorithm using those limits.

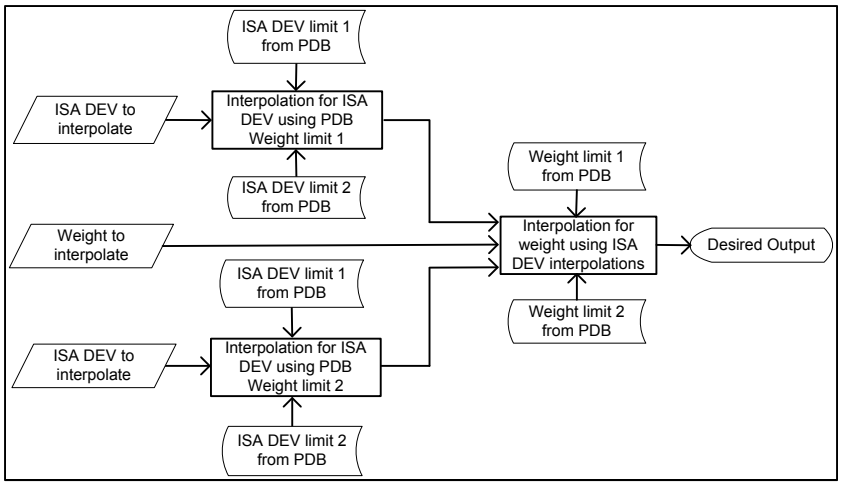

Figure 3.10 Interpolation path for a desired value

In Figure 3.10 is denoted that firstly the interpolations for the needed ISA deviation temperature are obtained, and then using those results, an interpolation for the weight is performed. For example for a flight in its CLIMB KIAS phase, flying at an hypothetical speed of 110 KIAS, with a ISA temperature deviation of 3.5 ºC, at 5,000 ft and a total weight of 127,500 kg. Using the PDB in Figure 3.9 and the interpolation path shown in Figure 3.10, the operations needed to obtain the fuel burned for this particular case are:

1. $f_1(ISA\ DEV\ required, Weight\ 1) = \frac{3.5-5}{0-5} \cdot 203 + \frac{3.5-0}{5-0} \cdot 209 = 207.2\ Kg$
2. $f_2(ISA\ DEV\ required, Weight\ 1) = \frac{3.5-5}{0-5} \cdot 216 + \frac{3.5-0}{5-0} \cdot 220 = 218.8\ Kg$
3. $f_3\big(Required\ Weight, (f_1, f_2)\big) = \frac{127500-130000}{120000-130000} \cdot 207.2 + \frac{127500-120000}{130000-120000} \cdot 218.8 = 215.9\ Kg$

The same logic is applied for the distance traveled.

CHAPTER 4

FLIGHT TRAJECTORY CALCULATION

Before describing the optimization procedure, it is important to describe the calculations and assumptions during each phase of flight. This chapter explains the way each phase of flight is calculated.

All the examples and graphs shown in this chapter make reference to the L-1011 aircraft. The calculations performed for this algorithm and the results presented later use the real data provided by CMC Electronics – Esterline. It is important to remind the reader that the PDB value steps such as KIAS, MACH number, weight, etc. can change from one aircraft to another and an analysis of the PDB has to be performed for each aircraft to implement the calculations exposed here.

During the trajectory calculation, 2two values are ultimately saved to finally calculate the cost of a flight: fuel consumption and flight time. During these calculations, the distance to travel is always known and the flight speed can always be calculated using equations (2.8), (2.9), and (3.6).

4.1 KIAS climb from 2,000 ft to 10,000 ft

All flights analyzed in this thesis begin at an initial geographical point at 2,000 ft of altitude and finish at the final geographical point at 2,000 ft of altitude. The reason of these limits is that there are many constraints depending on the airport exist and ATC for speeds lower than the first 2,000 ft. Another reason is that some PDBs do not have data available for altitudes below 2,000 ft. The default speed used to perform the calculation in this flight phase is 250 KIAS. This value can be changed if desired by the user, but it must not surpass 250KIAS due to the Code of Federal Regulations 91.117 [32]. Figure 4.1 shows the distance traveled against altitude for a given trajectory.

During the climb, the weight of the aircraft is updated at every 1,000 ft. This is because the PDB is divided in 1,000 ft multiple altitude. For example, if the aircraft began with a weight of 200,000 kg and a climbing from 2,000 to 3,000 ft, that required 300 kg of fuel, then the weight is updated to 299,700 kg. This means that the cost calculation to climb from 3,000 ft to 4,000 ft is calculated with the updated weight at 3,000 ft and so on.

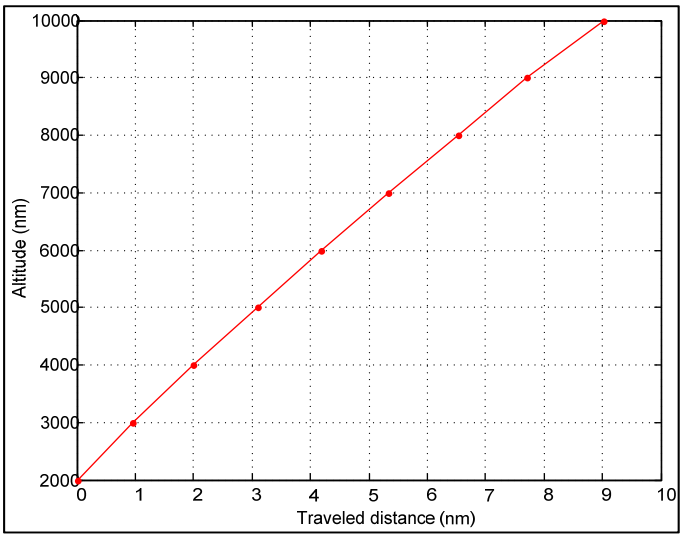

Figure 4.1 Distance traveled during the initial climb

4.2 Acceleration

After 10,000 ft, the climb continues in KIAS, but not necessarily at the same speed. Because of the way the PDB is arranged, during this phase, interpolations in speed are required. For the acceleration required, the PDB returns 3 outputs such as fuel burned, horizontal distance traveled and the needed altitude to arrive to the desired speed, this altitude is never a multiple of 1,000 ft. The acceleration phase then has two different stages: The first stage is computing the acceleration phase, and then the second stage is a small climb at the new constant KIAS needed to reach the next multiple of 1,000 ft altitude. The accelerations calculated in the algorithm are from the initial speed to the fastest KIAS available in the PDB.

Figure 4.2 represents the roadmap of the interpolations that need to be performed to complete the acceleration phase calculations. First, the delta speed that the aircraft has to accelerate is determined. Notice Figure 4.2 that there are two PDB initial speeds. Normally those initial speeds differ from 250 KIAS. Because of this difference, an interpolation of the delta speed needed is performed for the initial speeds available in the acceleration PDB. The results of those interpolations are then used to interpolate for the desired initial speed (normally 250 KIAS). These first interpolations are performed for the lower step weight. The same interpolations are performed for the highest weight step. Finally, an interpolation between the results of these weight values is performed to obtain the final results for the real weight of the aircraft at the beginning of the acceleration flight.

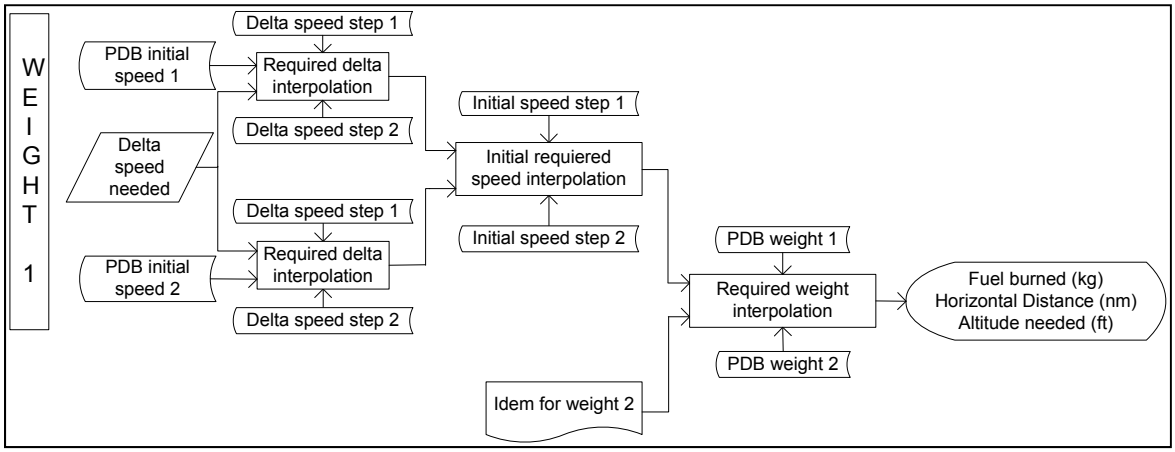

Figure 4.2 Acceleration interpolations path

After these interpolations, one of the output parameters is the altitude needed to perform the acceleration. This is where the second part of the acceleration phase calculations begins. The calculations performed in this algorithm have to be performed at altitude multiples of 1,000 ft. However, the needed altitude after acceleration calculated is never of a 1,000 ft. step. For example, Figure 4.3 shows an acceleration situation. Note how at the end of the acceleration phase the aircraft is located at an altitude of 12,520 ft. In order to reach the next available value in the database, the climb at the new speed has to be calculated from 12,520 ft. to 13,000 ft. But we have no way to access the altitude of 12,520 ft. from the PDB because it is not a multiple of 1,000. What it is done is that with the total weight of the aircraft after the

acceleration, a 1,000 ft climb from 12,000 ft. to 13,000 ft. is calculated and only 480 ft of that climb are considered. Equation 4.1 shows this type of interpolation.

$$fuel\ after\ acceleration = \frac{(A_1 - A_0)}{1{,}000} \cdot fuel \tag{4.1}$$

In this equation A_1 is the next multiple of 1,000 ft altitude, A_0 is the altitude after the acceleration and fuel is the fuel (kg) needed to climb from the last altitude multiply of 1,000 ft to the next altitude multiply of 1,000 ft. A similar equation is used to obtain the horizontal distance traveled.

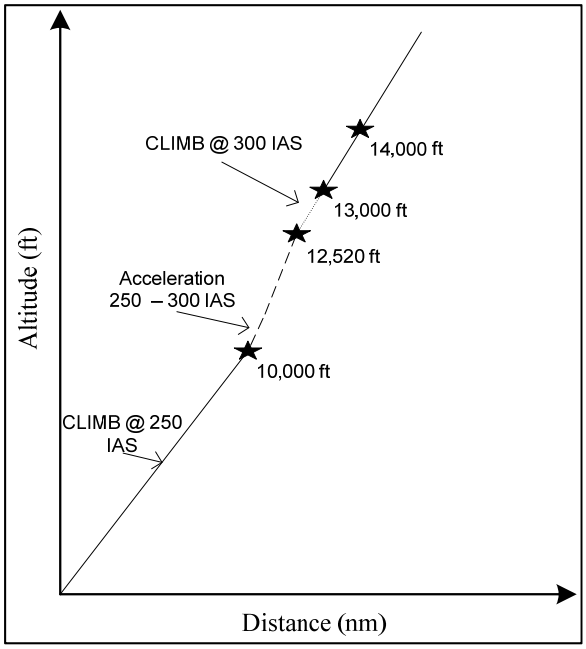

Figure 4.3 Acceleration during climb and
its constant climb acceleration

Finally, the cost obtained from the acceleration interpolation and the cost of the constant climb obtained in equation (4.1) are added together to find the cost for obtaining the final altitude of the airplane, the distance traveled and the fuel burned during the acceleration phase.

The speed used to calculate the flight time during the acceleration phase is the average of the speeds involved. For example, if the initial speed is 250 KIAS and the needed speed is 310 KIAS, the speed used to calculate the flight time is 280 KIAS. The small distance traveled after the acceleration is calculated with the final speed, 310 KIAS in this example.

4.3 Constant KIAS climb

In this phase, the climbs KIAS after the acceleration are calculated at a constant speed until the crossover altitude is reached. Similar as in Section 4.1, the weight is updated every 1,000 ft to have a more precise calculation. In this section the interpolations are again performed for the weight of the aircraft and the ISA temperature deviation standard. The typical interpolation process of the PDB is shown in Figure 3.10.

Table 4.1 Altitudes (ft) of some crossovers for different KIAS/MACH couples

250	260	270	280	290	300	KIAS/MACH
39,241	37,550	35,913	34,323	32,760	31,224	**0.81**
39,832	38,140	36,510	34,921	33,369	31,840	**0.82**
40,422	38,729	37,093	35,516	33,966	32,451	**0.83**

The calculation of the climb for the couple KIAS/MACH may lead to duplicate computations for the same climb. Caution has to be given in this part to avoid this and save calculation time. The crossover altitude for a given KIAS increases as the MACH number increases as it can be seen in Table 4.1. This means that, for example, for a KIAS of 270 the crossover altitude of MACH 0.83 is higher than the crossover altitude of MACH 0.82. Then if the climb 270/0.83 is calculated first, the pairs at 270/0.82, 270/0.81, 270/0.8, and so on, have the same values (fuel consumption and horizontal traveled distance) as the ones calculated with 270/0.83 until the crossover altitude is reached, after this altitude, the values will change. This can be seen in Figure 4.4 where 3 different climbs are calculated for 270/0.81, 270/0.82 and 270/0.83. The curves represent the variation of altitude with the distance in nm traveled for the 3 different pairs of KIAS/MACH. The first pair calculated was 270/0.83. Note how for the curve 270/0.82 all the points below the crossover altitude (36510 ft) are in

the exact same place as in the 270/0.83 curve. The same applies to the curve 270/0.81. Those last two curves values were not really calculated, but just copied from the 270/0.83 climb. Notice that the crossover altitudes in Figure 4.4 are added just as reference. The points are not located at exactly the crossover altitudes.

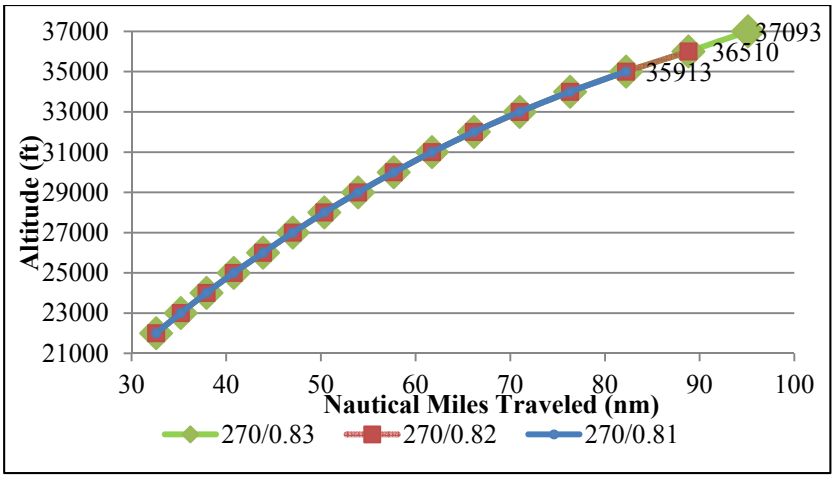

Figure 4.4 Traveled distance a a climb at 270 KIAS to 3 different crossovers

Some numerical values of Figure 4.4 can be found in Table 4.2. Here it can clearly be seen that if the computation of the climb 270/0.83 is performed first, the climb values of the pairs 270/0.82 and 270/0.81 can be just copied to the crossover altitude.

Table 4.2 Distance traveled in a KIAS climb at different crossover altitudes

KIAS	**270**		
Altitude(ft)/MACH	**Distance traveled (nm)**		
	0.81	**0.82**	**0.83**
31000	61.73	61.73	61.73
32000	66.16	66.16	66.16
33000	70.99	70.99	70.99
34000	76.32	76.32	76.32
35000	82.23	82.23	82.23
36000	*MACH*	88.82	88.82
37000	*MACH*	*MACH*	95.10

Please notice in Table 4.2 that the KIAS altitude for the pair 270/0.82 has values available up to 36,000 ft. However, Table 4.1 and Figure 4.4 show the crossover altitude to be at 36,510 ft. During the calculation at the KIAS climb, the algorithm will always stop to the multiple of 1,000 ft altitude just after the crossover altitude. This can be verified by comparing the numerical values shown in Table 4.2 with the crossover altitudes in Table 4.1.

4.4 Climb MACH

In the beginning of the climb MACH phase, the aircraft is located at the altitude multiple of 1,000 before the crossover altitude. Beginning to calculate from this point in MACH would give an error in the computations because there is one small segment that has to be calculated in KIAS. The crossover altitude is 36,510 ft. Therefore, it can be seen on Figure 4.4 at climb 270/0.82 that there are 510 ft while climbing from 36,000 ft to 37,000 ft in KIAS, and 490 ft after the crossover altitude (36,510 ft) that are calculated in MACH. During the first 1,000 ft of the calculation in the MACH climb phase, the 1,000 ft climb is calculated for KIAS and MACH. Equation (4.1) is used to calculate the influence of MACH climb during the first 1,000 ft of the MACH climb. Then by use of equation (4.1), the KIAS effect is calculated. Both values are added together and the first 1,000 ft fuel burned and horizontal distance traveled are obtained. After the first 1,000 ft of climbing in MACH after the crossover altitude, the rest of the computations is done in a normal way only in MACH until the maximum altitude is reached for the given aircraft configuration. Figure 4.5 shows the climb to many altitudes for different KIAS/MACH couples.

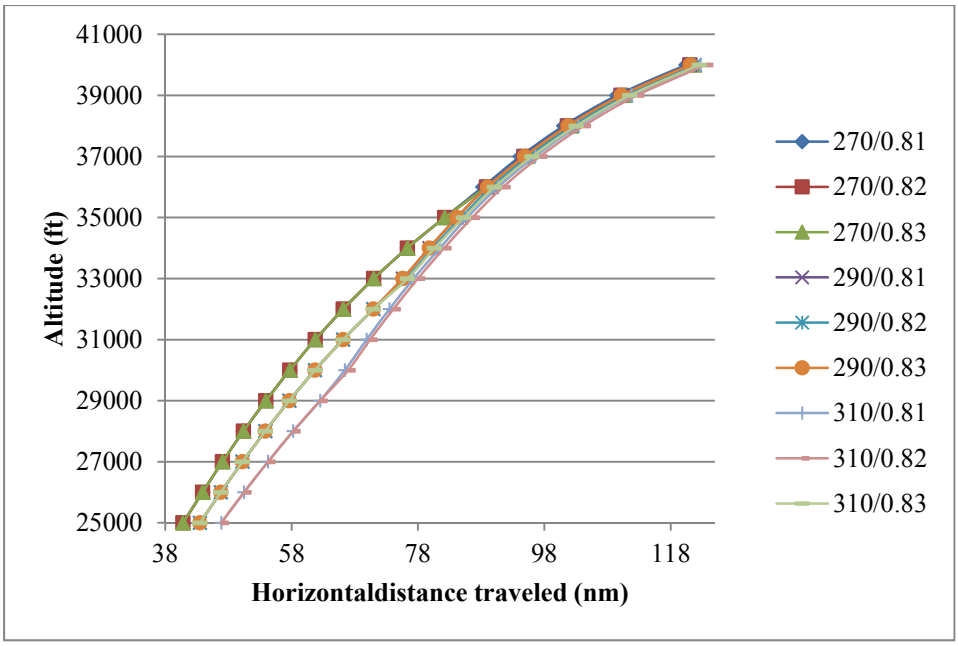

Figure 4.5 Horizontal distance traveled to many pairs KIAS/MACH

The algorithm calculates all the couples KIAS/MACH available during climb. To summarize, a block diagram is shown in Figure 4.6 that describes the climbing calculation algorithm.

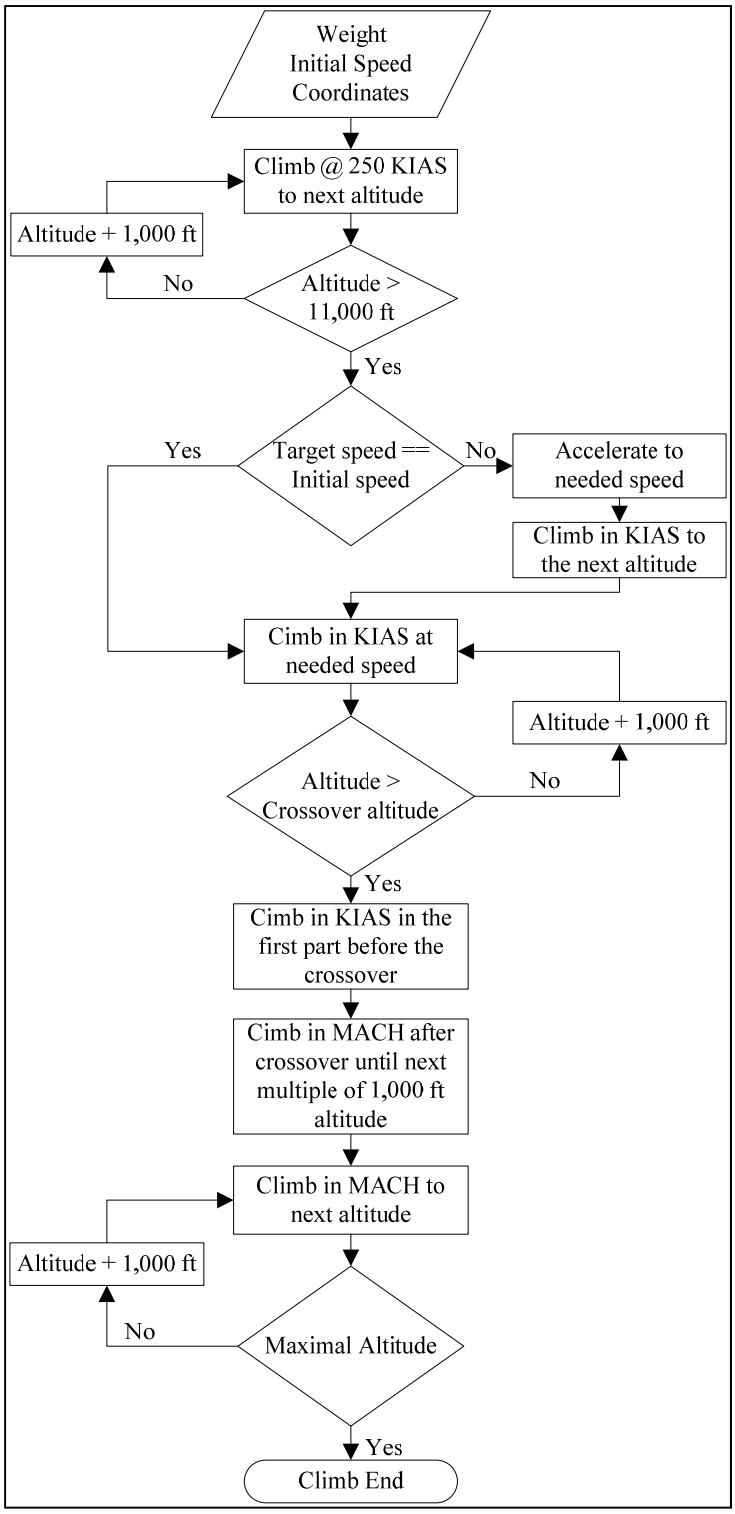

Figure 4.6 Climb computations flowchart

4.5 Descent distance estimation

Before calculating the cruise, an approximate descent distance has to be calculated in order to define an estimated Top of Descent (TOD). The estimation is performed by descending from an altitude, a MACH number and, an aircraft descent weight. The altitude, the MACH number and the descent weight used are those selected during a pre-cruise optimization method that will be explained in the next Chapter. The MACH/KIAS couple selected is the couple that takes the longest distance to descent, this was arbitrary selected. This distance is just a value estimated and a more precise descent calculus is explained and analyzed in Section 4.7.

4.6 Cruise

As it was stated in Section 2.1.2.1, the cruise is the distance between the TOC and the TOD. The algorithm is able to estimate all the TODs needed from the descent distance estimated. During this phase, the interpolation scheme is the one shown in Figure 3.8. However, during this phase the PDB does not provide the distance traveled by the aircraft, it has to be calculated by the algorithm using the direct and the inverse methods of Vicenty explained in Section 3.4 and detailed in Table 3.1. Starting at the TOC calculated during the cruise, the azimuth to the next point in cruise can be calculated with the Vicenty direct method. To calculate the next point in the cruise, the inverse method has to be used, but Vicenty's equations need to know the distance that the aircraft has to travel. The considered in these methods was 25 nm. When the aircraft travels this distance of 25 nm, the total weight of the aircraft is updated. This 25 nm distance was chosen because it is an acceptable compromise between computation resolution, error induced by not updating the weight of the aircraft instantly, size of the saved variables in the algorithm, computations resolution, and computational speed. Figure 4.7 is an analysis made with a 950 nm cruise at a constant altitude of 36,000 ft. The flight was performed 17 times, for every time the distance between points in the cruise was changed and the total fuel consumed is displayed. Note for the same flight how the fuel consumption augments for the same flight as the separation distance

between waypoints in cruise augments. That is the induced error due to the separation distance.

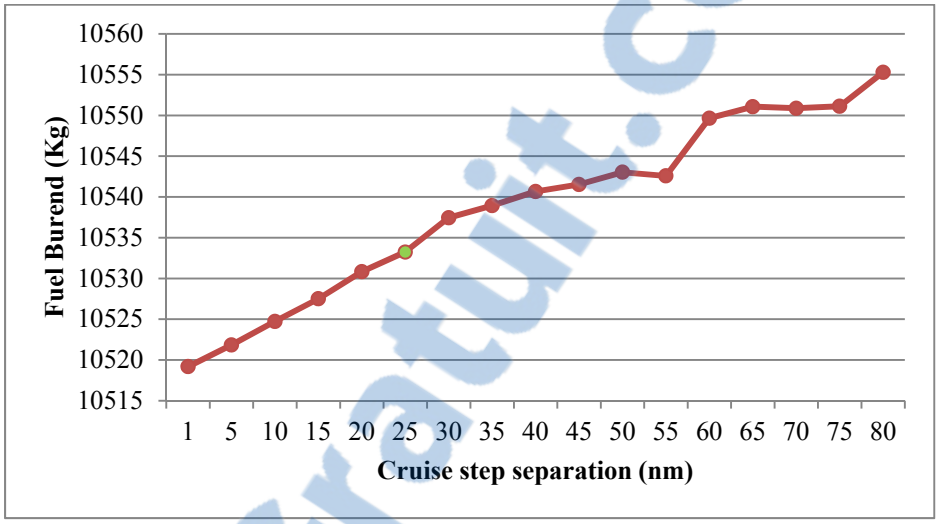

Figure 4.7 Fuel consumption change with different cruise separations

The induced error does not necessarily affect the optimal trajectory when the altitude remains constant during the flight. An extended explanation can be found in [16]. Nevertheless, when searching for step climbs, the precision of the calculation becomes important. The resolution to be chosen concerns directly the algorithm explained in this thesis because the "step climb search" is the method of savings chosen during cruise.

4.6.1 Step Climb

"Step climb" was selected to be implemented at every hour of flight in an effort to make this algorithm compatible with the standard ATC regulations [24]. An airplane cannot execute step climbs at if other airplanes are close to him. A permission of the ATC needs to be obtained before climbing to a different flight level (FL) using a step climb.

The algorithm identifies the geographical points of the aircraft's cruise trajectory that are next or exactly at every hour of flight. When the algorithm computes the cost of the current

altitude trajectory, the aircraft performs a small climb of 2,000 ft from the current altitude at the identified step climb and the rest of the cruise is performed at this new altitude by searching for more possible step climb opportunities. Finally Figure 4.8 shows the cruise calculation path.

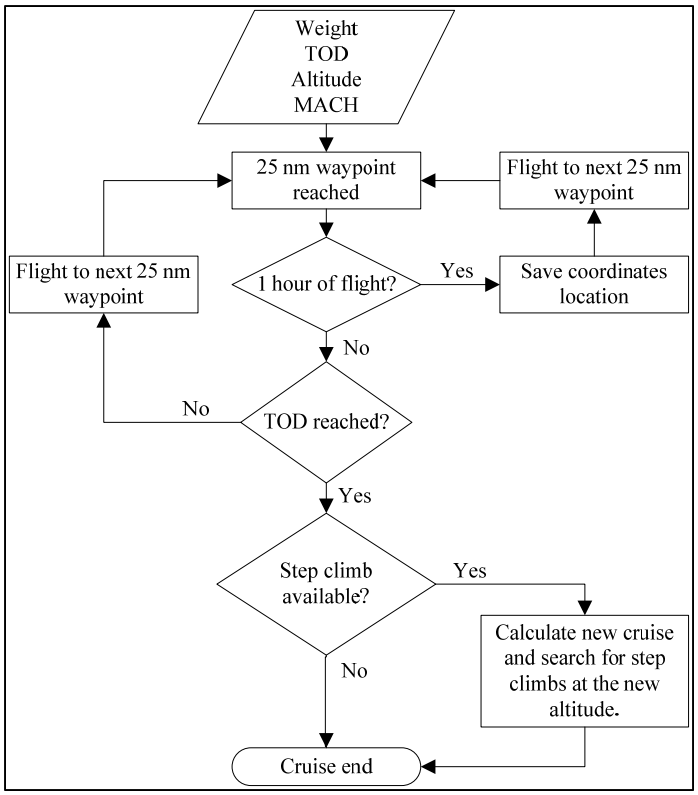

Figure 4.8 Cruise calculation path

4.7 Final descent

Final descent is highly dependent on the cruise. It was explained in Section 4.6 that the separation between waypoints during cruise is kept constant at 25 nm. This remains true normally until the last point before the estimated TOD. The algorithm verifies every 25 nm if the estimated TOD has not been surpassed. If the estimated TOD is surpassed, this exceeding distance is calculated and reduced from the default 25 nm and the final and more accurate descent phase is recalculated. After the calculation of the updated last cruise

distance, the airplane is located at the estimated TOD and the descent begins. Figure 4.9 describes this process.

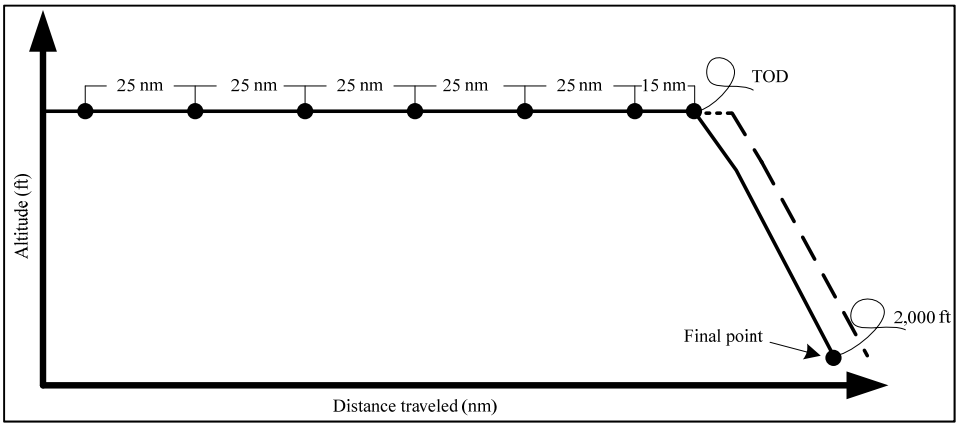

Figure 4.9 Cruise distance separation and descent correction

The descent looks like the KIAS climb phase; there is also a part of descent executed in MACH until the crossover altitude is reached. Similar as in KIAS climb, computations can be saved by copying the MACH calculations to the descent lower MACH/KIAS couples, in a similar way as it was done during Climb in KIAS in Section 4.3. After the crossover altitude, the descent in KIAS is calculated. The deceleration is needed to reach the 250 KIAS at 10,000 ft. Finally the descent is executed at this constant speed of 250 KIAS and ends at an altitude of 2,000 ft. The final location of the airplanes is then compared to the final point of the trajectory. If the final position of the aircraft is located after the destination point, or if it is not located within 500 meters before the destination point, the missing or surpassed distance is added or removed from the cruise phase, the TOD is modified and the descent is recalculated. This process is repeated until the aircraft ends within the imposed limits. Figure 4.10 is a description of the final descent procedure and the coupling with the cruise phase. The 500 meters distance was chosen because less than 500 m does not represent an important change in the total cost, and many factors can influence this distance such as weather influence and the pilot's skills.

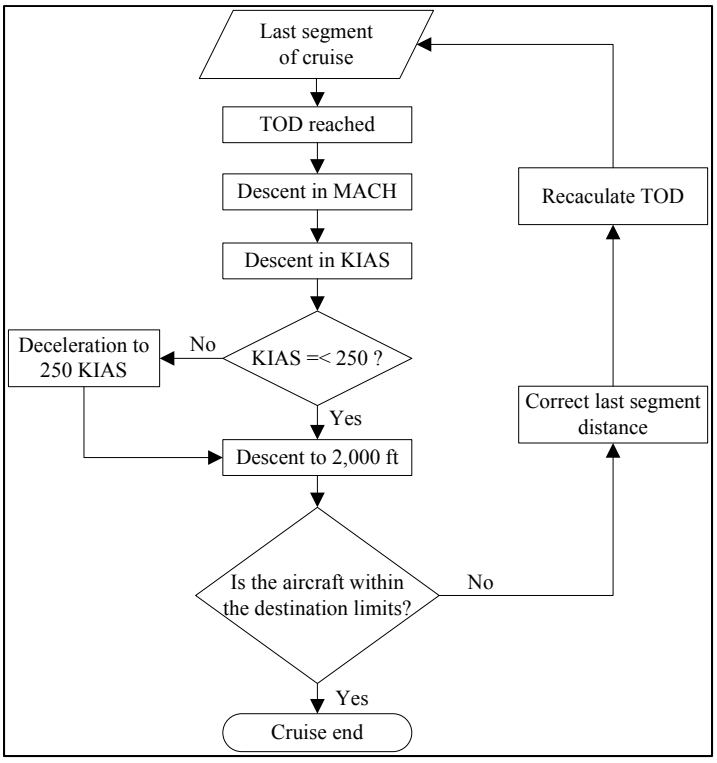

Figure 4.10 Descent phase calculation procedure

CHAPTER 5

TRAJECTORY OPTIMIZATION

In this chapter the way to determine the "optimal" Vertical Navigation (VNAV) is explained. Next the algorithm to find the "optimal" Lateral Navigation (LNAV) is exposed. Furthermore, the way of coupling these algorithms is explained to obtain the complete trajectory optimization.

5.1 Vertical navigation optimization

To find the optimal trajectory, flight trajectories with many speed/altitude combinations have to be calculated and compared. If "step climbs" are recommended they have to be calculated. Finally, all trajectories are compared and the one with the lowest cost is selected as the optimal VNAV trajectory.

The total number of calculations needed to find the optimal VNAV trajectory is high. For example, for the L-1011 during climb, 20 KIAS speeds can be found, 13 MACH and 11 cruise altitudes, that gives 2860 possible trajectories. The search for the optimal between all these combinations in a limited time frame is a difficult and time consuming task.

There are many ways to perform these calculations as it was explained in the literature review in Chapter 1. However, these methods are usually time-consuming. In order to find the optimal trajectory it will be efficient to reduce the number of optimal altitude/speed analyzed and its influence on the trajectory calculations. The reduction of MACH numbers and altitudes combinations will reduce calculation time. This time reduction allows the code implementation in the FMS. In order to reduce the number of combinations, a pre-optimal cruise optimization is performed, that is explained in the next section.

5.1.1 Pre-optimal cruise optimization algorithm

The idea of the pre-optimal cruise optimization algorithm is to try to have a first guess of the optimal cruise altitude/MACH profile. This first guess is defined as the "optimal candidate". This optimal candidate is the most important parameter in the algorithm because all the other calculations: climb, final cruise and descent explained in Chapter 4 are performed for the pre-cruise couple and in its vicinity. The advantage of applying the method describe here for the L-1011 PDB for example will be, that, when the climb will be computed, instead of searching a crossover altitude in the 13 possible MACH speeds, this crossover altitude will be searched 3 or 5 speeds in order to find the climb cost. It can be interpreted as a reduction of 8 MACH crossovers for each KIAS climb. It is evident that a reduction in the number of possible optimal solutions is observed.

During this phase, all MACH numbers and altitudes available in the PDB during the cruise phase are calculated and the total cost is found with equation (2.17). The least expensive cruise cost combination MACH/altitude pair is chosen as the "pre-optimal cruise pair".

In order to be able to use the pre-cruise optimization algorithm, a test trajectory has to be determined. The procedure to compute the test distance and the pre-optimal cruise pair is as it follows:

1) Calculate the distance (Di_0) between airport A and airport B.

2) Calculate the real climb cost from 2,000 to 10,000 ft at 250 KIAS. The fuel burned (W_1) as well as the horizontal distance traveled (Di_1) variables are saved.

3) Airplanes are designed to have an optimal MACH speed at a certain altitude. These variables have to be known or estimated in order to select their correct values for a climb. For the L-1011, 0.82 MACH is the designed cruise speed and 36,000 ft is a typical cruise altitude. With the maximum step of weight, being 210,000 kg for the L-1011, using the

PDB in mode climb MACH the fuel consumption (W_2) and the traveled distance (Di_2) are fetched and saved.

4) For the descent, the traveled distance data is fetched directly from the descent KIAS PDB table. The KIAS is arbitrary selected, for this work a descent from 36,000 ft at 300 KIAS was the information used to fetch the data. The weight used was the maximum weight allowed to a descent, 170,000 kg for the L-1011. The data saved is only the horizontal distance traveled during the descent (Di_3). Note that the weight for the descent is not fetched. This is because the weight at the descent is not necessary to calculate the test weight for the cruise. Only the weight at the beginning of cruise is needed.

5) The 3 saved distances (Di_1, Di_2, and Di_3) are added together. The added distances are then subtracted from the total distance (Di_0).

6) The saved weights (W_1 and W_2) are added for the 2 distances and then subtract them from the total weight (W_0), obtaining the test weight.

7) Calculate the cost of cruise for every MACH speed at every available using the cruise model described in Section 4.6 without calculating the descent and without evaluating the step climb and with only 8 waypoints during cruise regardless the flight distance. Only 8 points were chosen to assure a quick calculation and allow the possibility of add weather data.

8) Compare and find the least expensive and declare it as the pre-optimal cruise MACH/altitude profile.

Equations (5.1) and (5.2) can be written from the steps above to determine the weight and distance test. Where *Di test* is the test distance used to perform the cruise, *W test* is the weight used to perform the test and W_0 is the initial weight of the aircraft.

$$Di\ test = Di_0 - Di_1 - Di_2 - Di_3 \quad (5.1)$$

$$W\ test = W_0 - W_1 - W_2 \quad (5.2)$$

Figure 5.1 shows a graphical representation of the trajectory and the weight calculations.

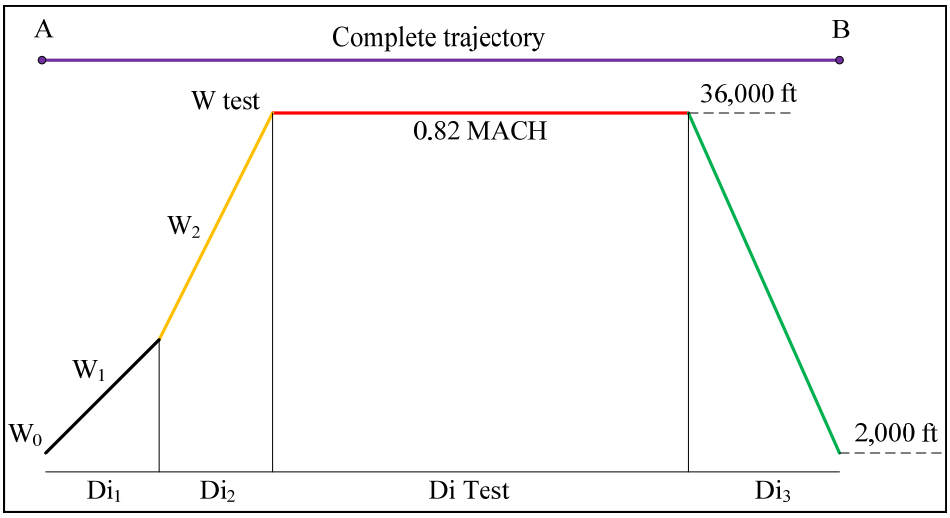

Figure 5.1 Pre-optimal cruise algorithm weights and distances

Figure 5.2 represents the results in terms of fuel burned (kg) of a pre-cruise evaluation. In this particular case it can be seen that the pre-optimal altitude is 36,000 ft. In this particular graph the MACH speed is impossible to be defined. Every asterisk represents a MACH.

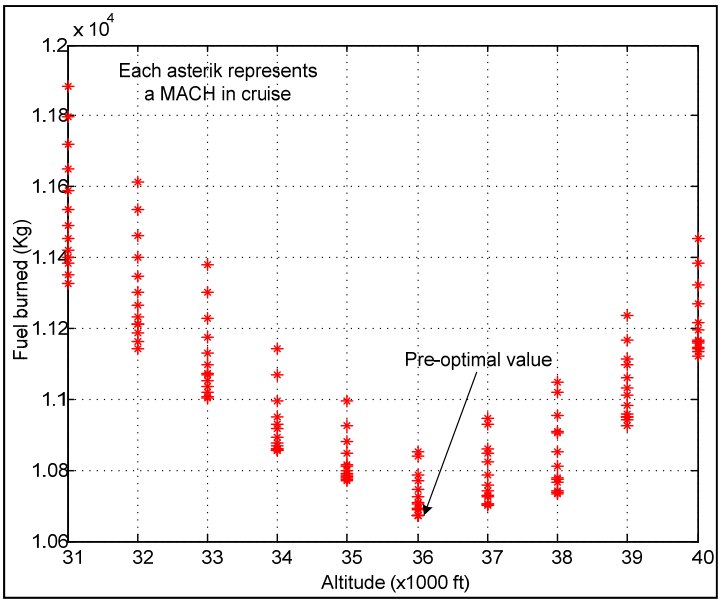

Figure 5.2 Pre-optimal cruise selection graph

Because the pre-cruise profile is obtained by focusing on the cruise, it does not give good results when it is used for short flights were the climb has a big influence in the flight's total cost. During a long cruise of around 700 nm and up, the climb cost is just a fraction of the total cost and, although a good climb has to be found, it does not affect the optimal MACH/altitude cruise during long flights.

5.1.2 Pre-optimal cruise results versus the algorithm of reference

In order to validate the pre-cruise initial solution, the algorithm was tested and compared with the algorithm developed by Gagné [18]. During this test, it was supposed that Gagné's algorithm provided the optimal cruise profile. The comparison tests are shown in Table 5.1. These tests were performed in a 1324 nm flight, the distance of a Los Angeles – Minneapolis, with different weights. This flight was chosen as a test because is a commercial flight with a mean distance of a commercial flights in North America. It can be seen that in almost all weights, the optimal profile solution during cruise found by Gagné's algorithm was exactly the same as our pre-cruise solution. Out of 9 cases, 3 cases highlighted in gray show different results.

Table 5.1: Comparison between Gagné's optimal versus pre-cruise first estimation

Distance: 1324 nm		Gagné optimal		Pre-cruise	
Test	**Weight(Kg)**	**Altitude (ft)**	**Speed (mach)**	**Altitude (ft)**	**Speed (mach)**
1	189000	34000	0.82	36000	0.82
2	185000	36000	0.82	36000	0.82
3	179000	36000	0.82	36000	0.82
4	174000	36000	0.82	36000	0.82
5	169000	36000	0.82	38000	0.82
6	164000	38000	0.82	38000	0.82
7	159000	38000	0.82	38000	0.82
8	154000	38000	0.82	40000	0.82
9	149000	40000	0.82	40000	0.82

Pre-optimal tests cruise profiles that are not exactly as the optimal ones given by Gagné's algorithm gave an error of 2,000 ft. This is the reason for which the algorithm calculates not only the pre-optimal candidate altitude and speed, but also computes and evaluates the altitudes lower and higher to the pre-optimal cruise. It is the same with speed, not only the pre-optimal speed is evaluated, but also the lower and the higher speed step values available in the PDB near the pre-optimal one.

5.1.3 Number of waypoints in the pre-optimal cruise algorithm

The last parameter to define for the pre-cruise is the number of waypoints where the weight has to be updated during cruise. As analyzed in Section 4.6, the waypoints separation during cruise has an effect on the resolution of the calculations.

The study performed to observe the variation of the final solution in terms of altitude and MACH with the variation of waypoints where the weight of the aircraft is updated was performed using the trajectory of LAX to MNP with the weight of 164,000 kg. The test was performed for a total of 11 different cruises that were created dividing them in 4, 6, 8, 10, 12, 14, 20, 40, 50, 60 and 100 waypoints. This means segments of 331 nautical miles (n = 4),

264.8 nm (n = 5), 165.5 nm (n =6) and so on. The expected result for this test is 38000/0.82. With those waypoints created, the pre-cruise function was executed and the pre-optimal profile in cruise was obtained; results are shown in Table 5.2 in terms of computational time and the optimal profile.

Table 5.2 Influence of the cruise computation resolution in the pre-optimal values

n	**Optimal Altitude (ft)**	**Optimal speed (MACH)**	**Computation time (s)**
4	38000	0.82	0.302506
6	38000	0.82	0.419559
8	38000	0.82	0.504747

Table 5.2 Influence of the cruise computation resolution in the pre-optimal values (continue)

10	38000	0.82	0.617521
12	38000	0.82	0.720401
14	38000	0.82	0.794593
20	38000	0.82	1.080818
40	38000	0.82	2.059276
50	38000	0.82	2.590815
60	38000	0.82	3.05723
100	38000	0.82	5.100623

As seen from the results, the computation time for an n = 100 (13.24 nm) was of more than 5 seconds, which is a considerable high calculation time for only an estimation. On the other hand, for an n = 4, the calculation time was only 0.3 seconds.

The conclusion of this study is that no matter the number of points in a cruise, the optimal cruise profile (altitude/MACH) was always the same one. The pre-cruise (altitude/MACH) results are independent of the number of waypoints chosen.

However, despite that logic suggests us to choose the lowest number of waypoints to reduce the calculation time, there is a different factor that is not considered in this study: the wind effect. As it was mentioned in the Section 3.5, wind changes along the route. Then, if the weather effects are to be taken into account, it is desirable to have the highest number of waypoints to get the most accurate wind variations, but then the computation time would be high. For this reason, the number of waypoints selected in this method is eight. This number of waypoints represents a good compromise between calculation time and accuracy for situations with wind effects and without wind effects.

As a final note, it is important to analyze the current meteorological information and the flight plan given from flight controls in order to choose the resolution value, eight points is a good approximation for the pre-optimal cruise flights in this work, but every day weather behaves in a different way, thus if a great variation of winds are seen in the weather chart, the number of points has to be increased in the algorithm to obtain a better prediction. If meteorological information is discarded, any number of waypoints can be chosen.

5.1.4 Climb and descent KIAS/MACH selection

After the selection of the pre-cruise speeds and altitudes, the climb computations are performed for all the available KIAS, and only 5 MACH: the pre-optimal MACH, the 2 before in the PDB and the 2 after the PDB. The maximal altitude calculated is the next one available in the PDB after the pre-optimal altitude found. As stated before, the number of combinations KIAS/MACH/altitudes and the calculation time are reduced. The results of these computations are tabulated in the algorithm. Tables contain the fuel consumption and the horizontal distance traveled are obtained and used within the algorithm. With these two tables, the ratio of nautical miles traveled per cost kilogram is calculated. Notice the term “cost kilogram”, this means that it is the cost defined by equation (2.17) where the cost index and the flight time are considered. The ratio with the maximum value is selected as the best climb profile.

For the descent, only the cruise MACH is analyzed. All the KIAS are calculated and once again the results are tabulated. The most economical MACH/KIAS descent is the profile selected as giving the optimal descent.

5.1.5 Step climb procedure and selection

The algorithm will execute step climbs during the cruise every hour to determine if flying at a different level would reduce the flight cost. The method of climbing to a different flight level was selected over the variation of the airspeed because during a flight, the commercial aircraft in flight are encouraged to maintain a constant airspeed, allowing ATC a better control of air traffic.

Many different algorithms have been developed at LARCASE searching for step climbs. These methods define waypoints, and every time a waypoint is reached a 1,000 ft, 2,000 ft or 4,000 ft step climb is analyzed. If the fuel flow at the new altitude is lower than at the actual altitude, those algorithms immediately suggest the aircraft to fly at a new altitude. These methods developed at LARCASE have given good results in fuel saving than the current FMS. Nevertheless, by climbing immediately and stop the analysis, the rest of the cruise leaves the uncertainty of not knowing if a different step climb performed later in the cruise gives better results because they are simply not analyzed. The “step climb” method proposed in this algorithm performs and computes all available “step climbs” indentified and decides which, from the available step climb locations is the best one to perform a “step climb”.

Figure 5.3 describes the process of computing the cost of a cruise, and shows the 4 possible VNAV trajectories for a given cruise. At the beginning, the cost of “trajectory 1” is calculated, at the same time, the algorithm tracks time and identifies the location of two possible step climbs points, called A and B. These points are identified at each hour of the duration of the flight. Once the cruise and the descent costs are calculated for this trajectory, the algorithm saves that trajectory profile and its cost in a table. The cost of “trajectory 1” includes the initial climb cost and descent cost. The algorithm calculates then “trajectory 2”

cost. It performs a step climb at point A and calculates the trajectory cost at the new altitude. Along the way, the algorithm identifies point C as a possible step climb. The cost of "trajectory 2" includes the costs of: the initial climb, the small cruise at altitude 1 from TOC to point A, the climb to a new altitude, the cruise at the new altitude, and the descent. When the cost calculation of "trajectory 2" is finished, the algorithm will save the profile and its cost in the same table used for "trajectory 1".

The algorithm proceeds then to calculate "trajectory 3" in a similar way as the way in which the "trajectory 2" was calculated. The cruise time from C to the TOD is lower than an hour, therefore no more step climb points are found. The results of "trajectory 3" are saved in the same table.

Finally the algorithm calculates "trajectory 4". It commands the aircraft to perform a step climb at point B. The flight cost is calculated and saved in a table. This process is repeated for all the altitudes and MACH defined during the pre-cruise phase of the algorithm.

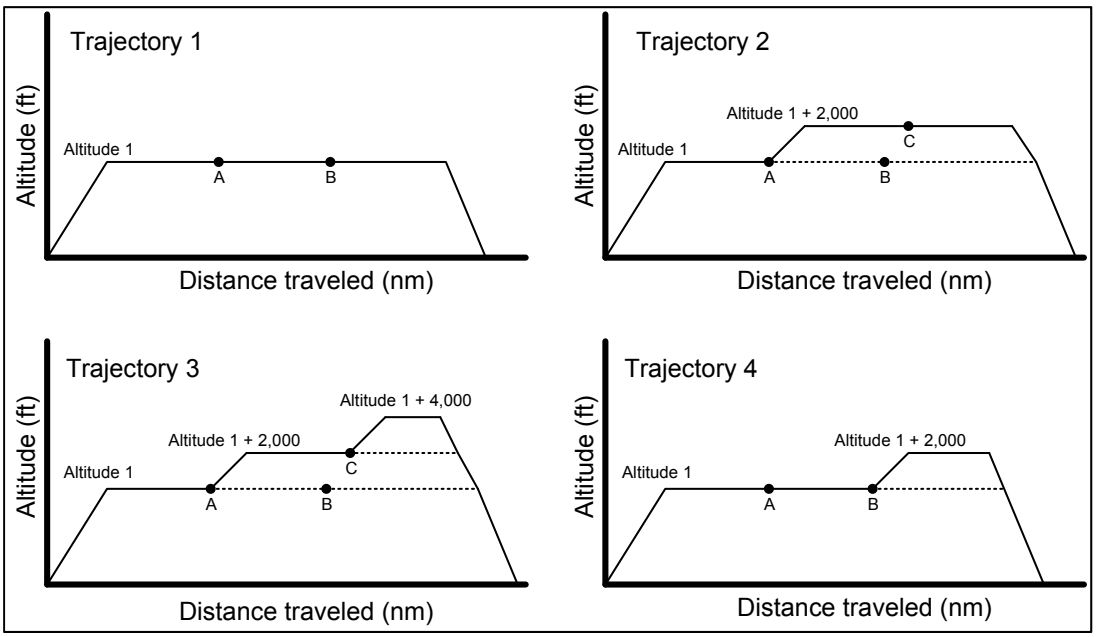

Figure 5.3 Trajectory options for a given altitude cruise analysis

Table 5.3 Costs of trajectories at a given altitude

Trajectory	Initial Altitude	Mach Speed	# Step Climbs	Final Altitude	Cost
1	Altitude 1	MACH 1	0	Altitude 1	Cost 1
2	Altitude 1	MACH 1	1	Altitude 1 + 2000 ft	Cost 2
3	Altitude 1	MACH 1	2	Altitude 1+ 4000 ft	Cost 3
4	Altitude 1	MACH 1	1	Altitude 1 +2000 ft	Cost 4

From Table 5.3, the algorithm would select and obtain the most economical trajectory for the profile altitude 1/MACH 1 in a new table that stores the profiles costs. The total cost includes climb, cruise and descent, and it considers the cost index influence.

5.1.6 Optimal VNAV route selection

Once all the possible cruises with their step climbs are calculated, the algorithm has all the information to determine the optimal trajectory. Notice that the cruise computations include already the costs of climb and final descent. Finally, as seen in Section 5.2.3, a flight cost table such as Table 5.4 for the cruise profiles defined in the pre-optimal cruise section is obtained. The least expensive trajectory is selected as the optimal trajectory. The climb, cruise and descent profiles are then displayed and deliver to the FMS to be followed.

Table 5.4 Final cost table

MACH/Altitude (ft)	**Pre-optimal altitude – 1 PDB index**	**Pre-optimal altitude**	**Pre-optimal altitude + 1 PDB index**
Pre-optimal MACH – 1 PDB index	Cost of profile $altitude_1/MACH_1$	Cost of profile $altitude_2/MACH_1$	Cost of profile $altitude_3/MACH_1$
Pre-optimal MACH	Cost of profile $altitude_1/MACH_2$	Cost of profile $altitude_2/MACH_2$	Cost of profile $altitude_3/MACH_2$
Pre-optimal MACH + 1 PDB index	Cost of profile $altitude_1/MACH_3$	Cost of profile $altitude_2/MACH_3$	Cost of profile $altitude_3/MACH_3$

Figure 5.4 is the flowchart of the different stages followed to determine the optimal trajectory by the algorithm.

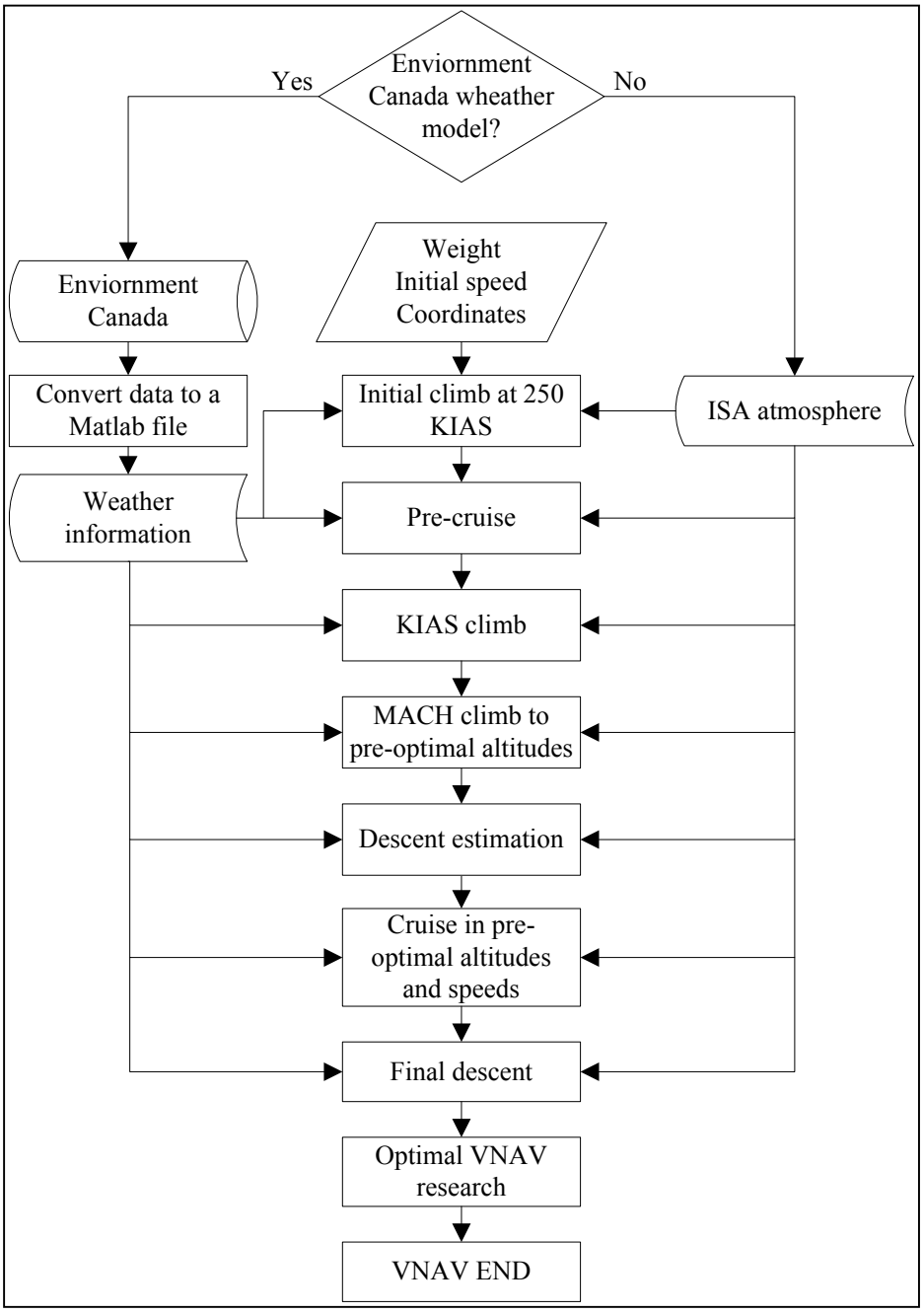

Figure 5.4 VNAV optimization path

5.2 Lateral Navigation Optimization

The coupling of VNAV and LNAV is a problem that has not been fully studied. In this Chapter, the second part of this algorithm, explains the way in which the VNAV is coupled with the LNAV. The meteorological information used is the one from Canada Environment described in Section 3.5.

5.2.1 Dijsktra's Algorithm

Gil in [27] developed at LARCASE an algorithm to search the optimal LNAV route by taking advantage of winds along the aircraft's path. A complete grid of waypoints was generated, and using the Dijsktra's algorithm the least expensive was selected as the optimal one.

The Dijkstra's algorithm found the shortest path between point A and point B. It calculated the cost in terms of flight duration between all the vertexes. Then, the costs of the available combinations were calculated. The least expensive route was defined as the optimal VNAV trajectory.

The algorithm developed by Gil had the disadvantages of computing the waypoints at only one altitude and optimization methods such as the "step climb" were not evaluated as their implementation would have been very difficult. The way in which the trajectories were created and how they have to be calculated made it hard to update the time the aircraft reached every waypoint. This forced the algorithm to assign meteorological conditions from Environment Canada using a constant time for all waypoints. It was as if a photo of the weather was taken and the meteorological information was obtained from that photo. One last problem identified with this algorithm was the effect that the grid of waypoints available was large and unnecessary. In a flight from Montreal to Paris, the grid could cover all the way to half Greenland and almost to the south of Portugal. The size of the grid and the number of waypoints available in that grid affected directly the computation time.

The first step to reduce the calculation time of this algorithm was to reduce the number of points. To achieve this reduction, the figure proposed by Gil was changed from a rhombus shape to a hexagonal shape. The number of waypoints available was reduced, thus the number of computations needed was reduced. The decision to obtain a hexagonal shape was taken after several tests shown that the extremes of the rhombus shape have never been found in the optimal trajectory, so it is fine to eliminate them. Figure 5.5 shows the difference between both shapes.

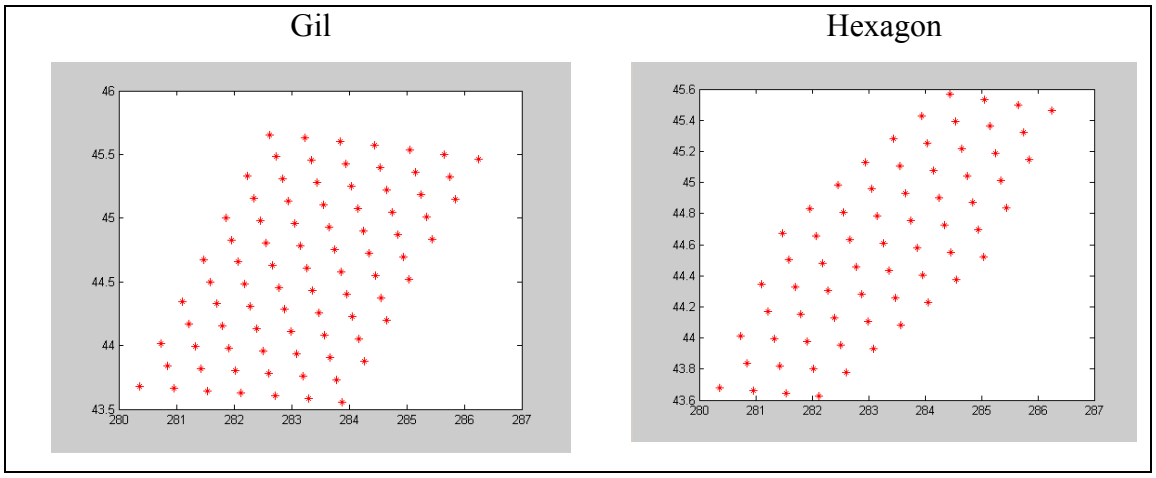

Figure 5.5 Gil shape versus hexagon shape

Tests and time measurements have shown the important reduction in calculation time by reducing the number of waypoints (vertexes) in Figure 5.5. Table 5.5 shows the obtained results.

Table 5.5 Comparison between Gil's shape and the hexagonal shape calculations

Flight: Montreal – Paris **Speed: 0.78 MACH**					
Algorithm	**Altitude**	**Sum Dir (hr)**	**Sum Op (hr)**	**Available vertexs**	**Calculation Time (s)**
Gil	30000	6.3177	6.3177	15000	0.043138
Hex		6.3177	6.3177	1933	0. 026837
Gil	35000	6.43	6.43	15000	0.04476
Hex		6.43	6.43	1933	0.027125

Table 5.5 Comparison between Gil's shape and the hexagonal shape calculations (continue)

Flight: Montreal – Paris **Speed: 0.78 MACH**					
Algorithm	**Altitude**	**Sum Dir (hr)**	**Sum Op (hr)**	**Available vertexs**	**Calculation Time (s)**
Gil	36000	6.45	6.45	15000	0.041830
Hex		6.45	6.45	1933	0.027978
Gil	37000	6.46	6.46	15000	0.043711
Hex		6.46	6.46	1933	0.026731
Gil	38000	6.489	6.489	15000	0.046369
Hex		6.489	6.489	1933	0.02676
Gil	39000	6.5052	6.5052	15000	0.043886
Hex		6.5052	6.5052	1933	0.026734

After multiple tests with the new trajectory shape, it was observed that the optimal route was in most of the cases the geodesic route. This is because the wind speeds and angles are similar for the other possible routes near the aircraft. If an important bigger tail wind current is found somewhat near the geodesic route, the aircraft spends more time and fuel arriving to that route. At the end of that route, the total cost for that route shows that it is better to maintain the geodesic one. In the few cases where the optimal route was different than the geodesic, the route identified was parallel to the geodesic.

It is also observed that this time saving during cruise using the Dijsktra's algorithm is mostly found in really long flights e.g. Montréal – Paris (2981 nm). In short and medium flights such as Montreal – Toronto (272 nm) and Los Angeles – Minneapolis (1324 nm), we can see different optimal routes than the geodesic one and the parallel ones. These different optimal routes can be found if the aircraft flies at very low speeds e.g. 0.3 MACH and at low altitudes e.g. 20,000 ft, this could be attributed to the effect of wind at low aircraft speeds. Normally, commercial airplanes such as the Sukhoi RRJ 100 and the L-1011 will not flight at such low speeds and altitudes.

Dijskstra's algorithm is a heavy algorithm in calculation time, and as it was shown before, if the geodesic route is the one that is the typical optimal one or parallel ones, this time expensed executing this algorithm in a device such as this FMS is not acceptable.

5.2.2 The five routes algorithm

Following the results observed from Dijskstra's algorithm, the five routes algorithm (5RA) method was developed. This method allows a fast LNAV calculation, it allows a time update to every point of the trajectory and procedures such as the "step climb" are easy to be implemented.

As mentioned above, typically the optimal route found is the geodesic route, since is the shortest, or a route parallel to the geodesic. This means that if 4 routes parallel to the geodesic route are created (two on the left and two on the right see Figure 5.6), is likely to find the least expensive route with minimal calculation efforts.

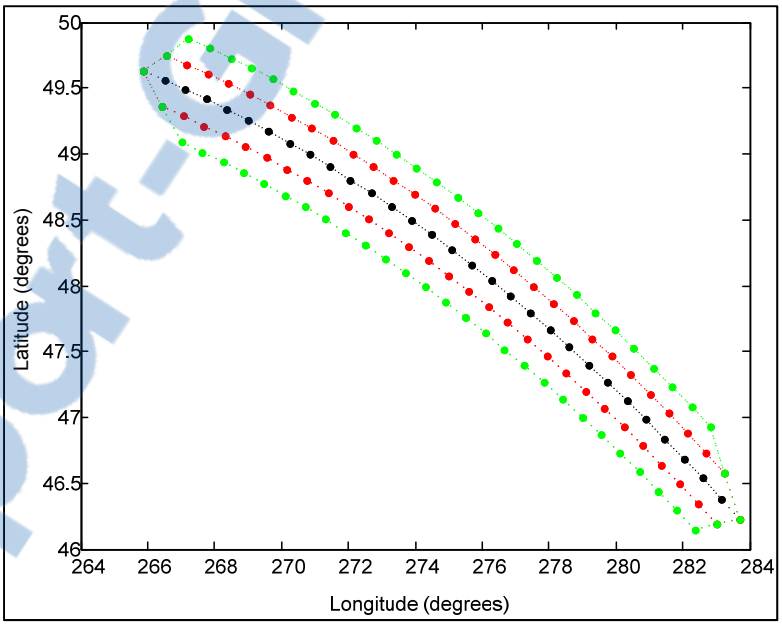

Figure 5.6 Five available lateral routes

The bad aspect of the 5RA is that an optimal route would have a pre-defined shape, if a "zig-zag" shape is the optimal route, it will not be possible to find it. Still these cases are very rare and can mostly be found in really long flights or at low speeds.

The capacity of this algorithm to update the time in every waypoint renders the calculation more precise since the exact meteorological information is taken into consideration. Table 5.6 shows the importance to update the time during flight by calculation flight duration.

Table 5.6 Flight time with static and dynamic weather

Altitude : 30000 ft Flight : YUL – PAR Speed: 0.8 Mach	Date : 30/march/2012 Time : 3:00 UTC	
	Static weather	Dynamic weather
Time (hr)	6.7664 hr	6.3249 hr
Delta dynamic – static (min)	**26.5 min**	

The five routes needed for this algorithm are created by defining the point of departure and the final points of the trajectory. For the 5RA defined in this work, the first route is the geodesic route known from the VNAV trajectory, for the other 4 parallel routes, the point of departure is the TOC and the final point is the TOD. From the TOC, a deviation angle is chosen and the next waypoint location at a distance of 25 nm is calculated. From the TOD, with the needed deviation angle a waypoint at 25 nm is also created in the direction of the initial point. Finally those points are connected with waypoints along the geodesic route of those two points created. A similar procedure is followed to create the other 3 routes.

5.2.3 Coupling VNAV with five routes algorithm

Once the VNAV trajectory is defined, the algorithm knows all the parameters needed to implement the five routes algorithm. These parameters are the TOC, the TOD, MACH and the altitude at which the airplane has to flight.

Knowing the TOC and the TOD, the algorithm knows the 4 parallel routes can be created. Because the geodesic route cost was calculated during the VNAV optimal calculations, there is no need to re-calculate the geodesic route. The parallel routes are the routes generated as it was explained in the last section and their costs need to be calculated in the same way in

which the cruise cost and descent are calculated. It means that for every route the TOD will change and the "step climb" is calculated to know if performing a step climb would economize the fuel cost. Figure 5.7 is a graphical description of the 5 routes algorithm, where the green route is the geodesic route.

After the calculation of the 5 routes, where the geodesic route is one of the 5 routes, the least expensive route is declared the "optimal lateral navigation route".

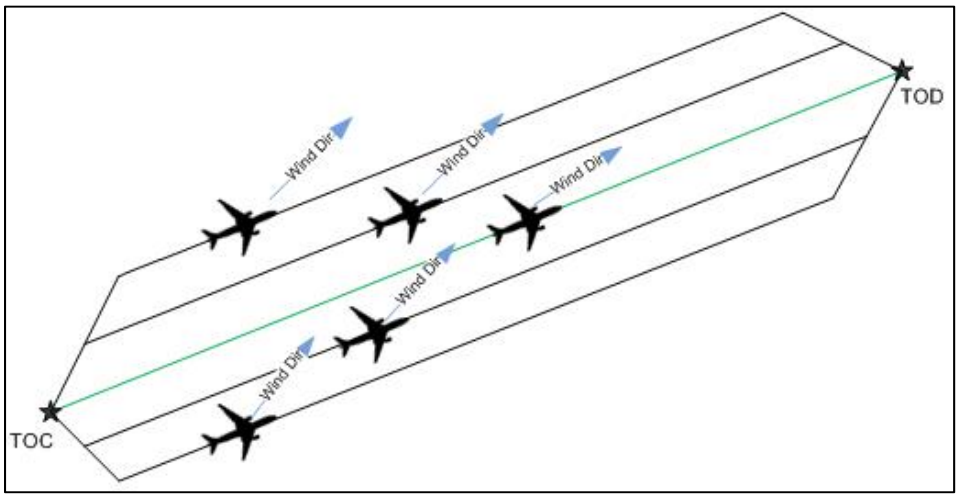

Figure 5.7 Five routes algorithm

As a final note, if the ISA model is used to define the weather, then the 5RA would not be calculated by the algorithm because the geodesic route would always be the optimal one. This is because the geodesic route is always shorter than the parallel routes, this longer distance causes the aircraft to consume more fuel.

CHAPTER 6

FUEL CONSUMPTION AND EMISSIONS GENERATED DURING A MISSED APPROACH

6.1 Introduction

Due to the proximity of airports to cities, very much attention has been given to the analysis and reduction of the following parameters: fuel burned emissions and noise produced in the landing phase of commercial aircraft as it was discussed in Chapter 1.

During the descent phase, many problems may appear that can lead to the landing procedure being aborted, in which case the so-called missed approach or go-around procedure must be followed. This procedure may be expensive. Flight crews can vary their approach procedures and flap selections to match the flight's objectives, which include fuel conservation, noise abatement and emissions reductions. Decisions on which type of approach to use vary with each airline, and sometimes even for each flight. The fuel required for a missed approach procedure during descent can burn up to 28 times the fuel consumed during a normal landing procedure [33] as specified in Boeing documentation.

Among the problems that can cause a landing procedure to be aborted are according to [34]:

- Unexpected traffic in the runway: Aircraft that are unable to take off on time and are still on the runway, aircraft flying close to the runaway, fast traffic overtaking the landing being performed, etc.

- Errors and misjudgments in the approach: Flying too high or too low on the final approach, flying too fast or too slow, overshooting the final approach start point, etc.

- Incorrect landing: Excessive bouncing at landing.

- Wind effects: A sudden change in the crosswinds, a wind direction or speed different than the expected wind direction and speed, problems with the automatic weather broadcast, etc.

Calculating the cost of a missed approach would be helpful to optimize aircraft systems such as the FMS in order to achieve and integrate a missed approach procedure. Three methods to calculate the emissions including one way to calculate the fuel burnt in any flight are described in [35]. However, these calculations are only performed for entire flights and not to specific landing approach procedures. These methods use information from the tables provided by the emission inventory guidebook (EIG) [35]. There is not too much other bibliographical research available in the field regarding the missed approach procedures evaluation.

The method described here calculates separately the fuel and the emissions spent in a missed approach procedure and those spent in a successful landing. These values are then added to the whole flight cost to compare the final cost of the missed approach procedure versus a flight without a missed approach procedure. This method separates the flight into two modes: one that is above 3,000 ft and the other below 3,000 ft. When the airplane flies below 3,000 ft, the time flown in this mode is calculated, and when the airplane is flying above 3,000 ft, the distance traveled is the value calculated. Once those two parameters have been obtained (distance and time), the distance is interpolated using the data in the EIG [35] to calculate the parameters of interest such as the fuel consumed, nitric oxide produced, etc. The time is multiplied for some conversion values, which will be explained later in this chapter to calculate the same parameters of interest mentioned above.

The new methodology described in this thesis for the missed approach procedure was implemented using Microsoft Excel 2007, Visual Basic for Applications (VBA) and Matlab.

6.2 Methodology

The new methodology described here is utilized for the two main modes: Climb/Cruise/Descent (CCD) and Landing to Take Off (LTO). The waypoints of the landing procedure and the waypoints of the missed approach must be defined. The waypoints are given in the instrument approach procedure charts, which the pilot should know in order to

perform the selected landing sequence. See Figure 6.1 for an example of an Instrument Approach Procedure (IAP) chart obtained from [36]. Each waypoint must have the correct information: altitude, flight speed, and distance between the current waypoint and the next one. Weather influence is not taken in consideration.

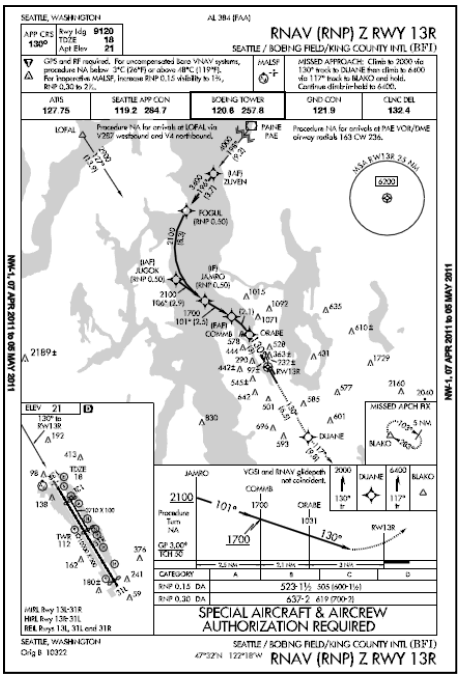

Figure 6.1 Instrument approach procedure chart
Source: Skyvector

By definition, the CCD and the LTO modes are separated at the level of 3,000 ft in altitude. The airplane is in the CCD mode when it is located at an altitude above or equal to 3,000 ft. If the airplane is located at another waypoint below 3,000 ft, then it means that is in the LTO mode [35].

The procedure of a missed approach is the following: The airplane starts to fly in the CCD mode at the initial waypoint (WPT) of the descent procedure found in the approach plate. It starts to descend, passing through all the WPTs in the CCD mode, eventually reaching 3,000 ft; below this altitude, the airplane enters into the LTO mode and continues to descend until it

reaches the decision point. If the pilot judges that everything is fine, the airplane lands and the flight ends. When the airplane is at the decision point, the pilot could decide to perform a missed approach procedure instead of landing, or Air Traffic Control (ATC) could command the pilot to perform the missed approach procedure; in which case new steps are followed to execute the missed approach procedure.

In the first step, the pilot activates the engines to perform the Take Off Go Around (TOGA), which means that the engines will be at their maximum power for the airplane to gain altitude and arrive to the cruise phase. The airplane will then follow different WPTs in the LTO mode, with the engines in a normal operation mode. Eventually the airplane will come back to the CCD mode, until it reaches a safe holding zone. Then, when traffic conditions allow it, ATC will assign the airplane a return vector and the pilot will try to land again. The airplane will follow this vector, which is normally within the limits of the CCD mode. Eventually the airplane will enter the LTO mode again to begin the landing approach. Figure 6.2 shows these procedures in a graphical form.

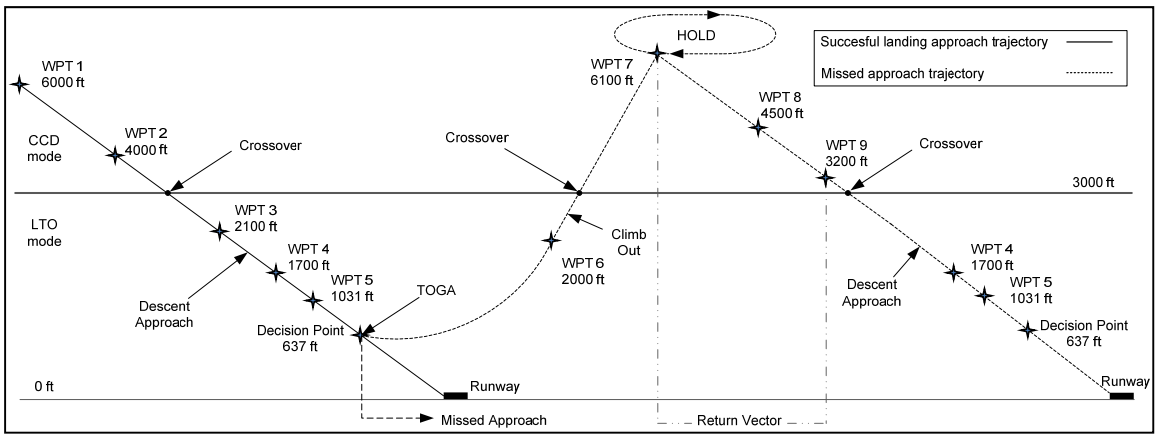

Figure 6.2 A successful approach and landing with a missed approach procedure

6.2.1 Climb/Cruise/Descent CCD mode

In order to perform the required calculations in the CCD mode, the total distance traveled in this mode must first be determined. This can be accomplished by verifying all the

consecutive WPTs of the defined trajectory and then calculating the CCD distance travelled. If two consecutive WPTs exist within the CCD mode limits, then the distance value is saved as an accumulative variable. This variable, which will eventually contain the total distance traveled in CCD mode, will be used to calculate the emissions of interest such as fuel, NOx, , HC, the Emissions Index of HydroCarbon (EICH), and the Carbon Monoxide (CO).

It is important to note that not all of the consecutive WPTs will be in the CCD mode -- some points will be located in the LTO mode, a situation which will be explained later in section 2.2 and 2.3. If all the WPTs are within the CCD limits, the total distance travelled in the CCD mode can also be expressed in the following way:

$$\text{Total distance in CCD} = \text{WPT 1 to WPT 2 distance} + \text{WPT 2 to WPT 3 distance} + \text{WPT 3 to WPT 4 distance} + \ldots \quad (6.1)$$

Depending on the total nautical miles (nm) traveled in this mode, an interpolation or an extrapolation of the distance may be needed. If the CCD distance is greater than 125 nm, an interpolation of the distance in the tables provided by the EIG is required [8]. This distance of 125 nm was chosen as the lower limit because it is the smallest distance in CCD mode given in the EIG consumption tables [35].

Nevertheless, the distance traveled in the landing approach procedure in the CCD mode is usually less than 125 nm. Since the EIG tables do not have values below 125 nm, a vector must be created, for which the first distance is not 125 nm but 0 nm, and the fuel consumption at 0 nm is considered to be 0 kilograms (kg). This vector makes it possible to interpolate from 0 nm to the maximal distance value available in the EIG tables for a specific aircraft. As an example, for the values of the distance and fuel consumption parameters of the Boeing 737-400, our vectors of distance and fuel are represented in the next two equations:

$$\text{Distance (nm)} = [0 \;\; 125 \;\; 250 \;\; 500 \;\; 750 \;\; 1000 \;\; 1500 \;\; 2000] \quad (6.2)$$

$$\text{Fuel consumption (kg)} = [0 \;\; 777.7 \;\; 1442.6 \;\; 2787.4 \;\; 4134.9 \;\; 5477.2 \;\; 8362.3 \;\; 11342.2] \tag{6.3}$$

With these vectors, a polynomial of interpolation of a given order can be used to calculate the fuel consumption as a function of distance. In this work, the polynomial of order 7 was selected because 8 was the lowest order where the real values were almost the same as the interpolated values. In Figure 6.3, the polynomial function is traced versus the real data expressed by equations (6.2) and (6.3). It can be seen that there is no error, because the polynomial function superposes over the real data.

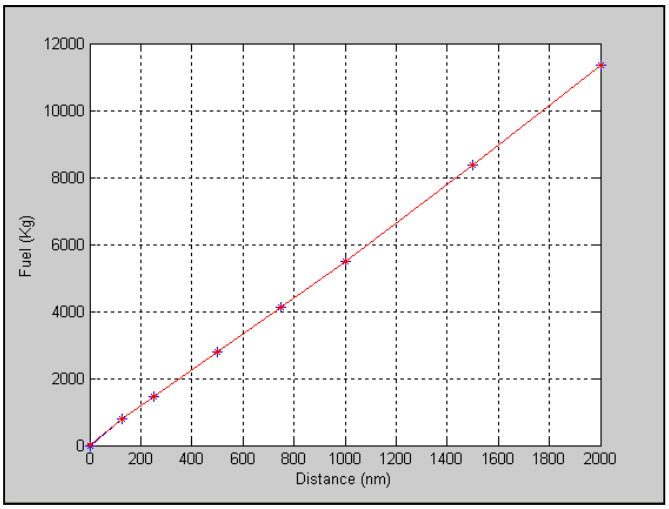

Figure 6.3 Polynomial interpolation function versus real data

To find the values of polynomial coefficients for the fuel consumption as a function of the distance x, the Matlab function *polyfit* was applied. The resultant polynomial Fuel(x) for the Boeing 737-400 is shown below:

$$\text{Fuel}(x) = 3.1721^{-18}x^7 - 1.9907^{-14}x^6 + 4.8834^{-11}x^5 - 5.9714^{-8}x^4 + 3.8605^{-5}x^3 - 0.0128x^2 + 7.3288x + 0.613^{-11} \tag{6.4}$$

The fuel consumption dependency with the distance is obtained by solving the polynomial function expressed in eq (7.4) for the distance x value; the fuel (kg) was thus calculated for the CCD mode.

Polynomials of order 8 are found for each parameter of interest: Fuel, NOx, HC, and CO. The polynomials created for the variation of these parameters with distance for the Boeing 737-400 are described by the following equations:

$$No_x(x) = 1.8042^{-18}x^7 - 1.0821^{-15}x^6 + 2.5081^{-12}x^5 - 2.8556^{-9}x^4 + 1.6815^{-6}x^3 - 4.9689^{-4}x^2 + 0.1165x + 0.468^{-12} \quad (6.5)$$

$$HC(x) = -1.7991^{-18}x^7 + 1.011^{-14}x^6 - 2.1141^{-11}x^5 + 1.9988^{-8}x^4 - 7.4545^{-6}x^3 - 6.1818^{-4}x^2 + 1.3658x - 3.5121^{-11} \quad (6.6)$$

$$CO(x) = -2.3525^{-17}x^7 + 1.3095^{-13}x^6 - 2.694^{-10}x^5 + 2.4633^{-7}x^4 - 8.2008^{-5}x^3 - 1.6059^{-2}x^2 + 22.2419x - 4.3744^{-10} \quad (6.7)$$

Where x is the distance expressed in nautical miles.

6.2.2 Landing to Takeoff LTO mode

In the LTO mode, three different phases can be identified: Approach Landing, Climb Out and Takeoff Go Around or TOGA. All these phases occur during a missed approach procedure. According to the ICAO, the three phases of an LTO mode have fixed reference times [35] that are shown in Table 6.1. The fuel and emissions calculations in these reference times are provided in the EIG tables [35].

Table 6.1 ICAO reference times

LTO PHASE	REFERENCE TIME (min)
Approach Landing	4
Climb Out	2.2
Takeoff /TOGA	0.7

In order to identify in which phase an airplane is located between two WPTs, the following definitions of these phases must be considered:

- *Approach Landing*: The WPT (*n*-1) is at the same or higher altitude than the WPT (*n*).
- *Climb Out*: The WPT (*n*-1) is at a lower altitude than the WPT (*n*).
- *Takeoff Go Around TOGA*: At the Decision Altitude (DA), the pilot determines if the landing procedure should be aborted, therefore the missed approach procedure should start.

In a successful approach, where no missed approach procedure is executed, the Approach Landing phase is the only phase considered for the LTO calculation, since in a successful landing, the aircraft is not supposed to climb or to TOGA. If an airplane needs to climb, that indicates that either the aircraft is leaving airport A on its way to airport B or that the airplane is performing a missed approach procedure.

After the LTO mode has been determined, the traveled time is calculated using the speed of the airplane and the distance between WPTs. This time is saved as an independent accumulation variable for two LTO phases: approach landing and climb out. The Takeoff/TOGA time value is always considered to be 0.7 minutes; this is because in this acceleration phase is complicated to determine the exact speed of the aircraft. The time saved in the accumulation variables is multiplied by the parameter of interest per minute, determined from the tables using the ICAO reference times above-mentioned in Table 6.1, thereby providing the fuel consumption. It is important to note that the EIG tables have

different parameters such as fuel, NOx, HC, CO, etc., and these calculations apply to all of them. One type of EIG table from [35] is shown in Table 6.2 for the B-737-400.

As an example, the quantity of fuel burnt in an Approach Landing phase of 6 minutes is calculated in the following way: According with Table 6.1, the reference time for an Approach Landing phase is 4 minutes. Looking at Table 6.2, regardless the flight distance, the fuel burnt during the Approach Landing phase is 147.3 kg for this particular aircraft. By dividing 147.3 kg between 4 minutes, it can be found that the aircraft burns 36.825 kg. of fuel per minute during this phase. Thus, the quantity of fuel burnt in 6 minutes during the Approach Landing phase can be found by multiplying the quantity of fuel burnt times the total time in this phase. This is 6 minutes times 36.825 kg/min. This results in a total of 220.95 kg.

Table 6.2 EIG table for the Boeing 737-400

B737 400	Standard flight distances (nm) [1nm = 1.852 km]					
	125	**250**	**500**	**750**	**1000**	**1500**
Distance (km)						
Climb/cruise/descent	231.5	463	926	1389	1852	2778
Fuel (kg)						
Flight total	1603.1	2268.0	3612.8	4960.3	6302.6	9187.7
LTO	825.4	825.4	825.4	825.4	825.4	825.4
Taxi out	183.5	183.5	183.5	183.5	183.5	183.5
Take off	86.0	86.0	86.0	86.0	86.0	86.0
Climb out	225.0	225.0	225.0	225.0	225.0	225.0
Climb/cruise/descent	777.7	1442.6	2787.4	4134.9	5477.2	8362.3
Approach landing	147.3	147.3	147.3	147.3	147.3	147.3
Taxi in	183.5	183.5	183.5	183.5	183.5	183.5
NOx (kg)						
Flight total	17.7	23.6	36.9	48.7	60.2	86.3
LTO	8.3	8.3	8.3	8.3	8.3	8.3
Taxi out	0.784	0.784	0.784	0.784	0.784	0.784
Take off	1.591	1.591	1.591	1.591	1.591	1.591
Climb out	3.855	3.855	3.855	3.855	3.855	3.855
Climb/cruise/descent	9.462	15.392	28.635	40.425	51.952	78.047
Approach landing	1.240	1.240	1.240	1.240	1.240	1.240

Table 6.2 EIG table for the Boeing 737-400 (continue)

Taxi in	0.784	0.784	0.784	0.784	0.784	0.784
HC (g)						
Flight total	817.6	912.9	995.8	1065.2	1118.1	1240.4
LTO	666.8	666.8	666.8	666.8	666.8	666.8
Taxi out	321.18	321.18	321.18	321.18	321.18	321.18
Take off	3.09	3.09	3.09	3.09	3.09	3.09
Climb out	10.58	10.58	10.58	10.58	10.58	10.58
Climb/cruise/descent	150.78	246.13	329.05	398.47	451.33	573.67
Approach landing	10.74	10.74	10.74	10.74	10.74	10.74
Taxi in	321.18	321.18	321.18	321.18	321.18	321.18
CO (g)						
Flight total	14252.5	15836.0	17525.5	19060.6	20369.3	23298.2
LTO	11830.9	11830.9	11830.9	11830.9	11830.9	11830.9
Taxi out	5525.45	5525.45	5525.45	5525.45	5525.45	5525.45
Take off	77.19	77.19	77.19	77.19	77.19	77.19
Climb out	202.29	202.29	202.29	202.29	202.29	202.29
Climb/cruise/descent	2421.54	4005.06	5694.59	7229.65	8538.39	11467.26
Approach landing	500.54	500.54	500.54	500.54	500.54	500.54
Taxi in	5525.45	5525.45	5525.45	5525.45	5525.45	5525.45

6.2.3 Crossover calculations

It is important to point out that the crossover concept does not refer to the typical KIAS/MACH speed change due to the altitude, in which the pilot has to change the speedometer from KIAS reference to MACH reference. The crossover in this procedure refers to the changes in the 3,000 ft threshold that separates LTO mode from CCD mode. During the missed approach, there are at least three crossover situations referring to:

1. The zone below 3,000 ft where descent is performed prior to landing;
2. Climb out over 3,000 ft after the abortion of landing at the decision altitude to wait in the holding pattern for a returning vector to land; and
3. Descent again below 3,000 ft in order to land after coming back from the holding pattern.

It is important to separate the CCD and the LTO modes during all crossover situations, such as those pointed out above.

A constant slope is considered in the descending or ascending path followed during the crossover situation. When the airplane is at the initial distance Dist (*n*-1), it is at the WPT (*n* - 1) at altitude ALTWPT (*n* - 1). At the final distance Dist (*n*), the airplane is at the WPT (*n*) at the altitude ALTWPT (*n*). Figure 6.4 shows the variation of the altitude with the distance while descending.

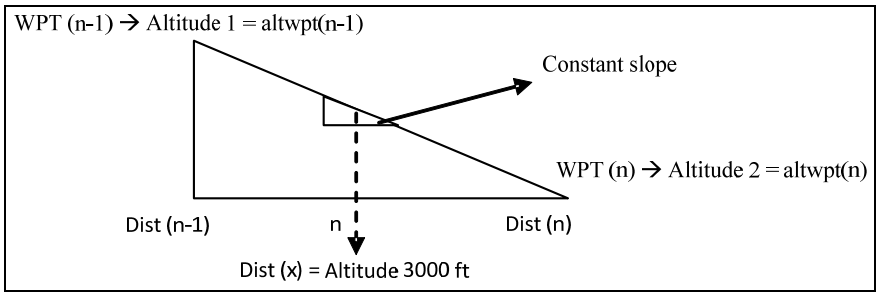

Figure 6.4 Altitude variation with distance

$$\text{Distance to reach } 3000\ ft = \frac{(3000ft - altwpt(n))(altwpt(n-1) - altwpt(n))}{distance\ between\ wpts} \qquad (7.8)$$

The number of nautical miles (*nm*) needed to travel from the current altitude (ALTWPT (*n* - 1)) at the WPT (*n* - 1) to arrive at the crossover altitude in the LTO or CCD mode is determined using eq. (6.8). If we are descending, we find the *nm* in the CCD mode, and if we are climbing out, we find the *nm* in the LTO mode. Note that equation (6.8) is valid when the altitude of WPT (*n*) is lower than the altitude of WPT (*n*-1). A similar analysis is done in the case where the WPT (*n*) is higher than the altitude of WPT (*n*-1).

6.2.4 Full flight cost calculation

In order to calculate a missed approach's additional cost with respect to the entire flight's cost, we need to consider the distance of a hypothetical full flight route from the end of the LTO mode during taking off from Airport A to the first waypoint of the landing procedure at Airport B. Using the distance of the hypothetical full flight route, the polynomials (6.4) – (6.7) can be solved for this particular case in order to find the parameter(s) of interest.

To calculate the LTO modes during takeoff, the values of interest contained in the EIG tables [35] and shown above in Table 6.2 are taken directly from these tables, assuming the standard reference times given by ICAO [35] and shown above in Table 6.1. The takeoff and climb out phases are accounted for and it is assumed that the fuel and emissions values in the EIG are the values of this LTO mode. Once the distance of the full flight route cost has been determined and the LTO cost obtained, these costs are added together to obtain the full flight cost for each single parameter of interest.

Note that the approach landing phase in the takeoff section is assumed to be equal to zero (0). This value is zero because when the airplane is leaving an airport, it is not supposed to descend at all, as that would mean the airplane is arriving at the airport instead of leaving it.
In the CCD mode, three different flight phases are combined into one phase: climb, cruise and descent. In a normal flight, during the climb phase more fuel is spent than in cruise mode, and in cruise mode, more fuel is spent than in descent. The method presented here does not take into account the differences between these varying levels of consumption and assumes that all three phases considered in the CCD mode comprise one big phase that is only a function of distance. These differences are not considered here because of the lack of that type of detailed data in the literature.

CHAPTER 7

ALGORITHM RESULTS

For this work, many tests were performed to validate the calculations and results of the algorithm. First, calculation validity tests were performed. Five flights tests were performed with the algorithm developed in this work and with FlightSim by Presagis, then the results were compared to verify that the trajectories calculated were correct. The exact cost of a given flight was obtained using FlightSim, this software is the same one as the one that CMC Electronics – Esterline uses to validate their algorithms. Secondly, comparisons of VNAV trajectories costs were performed between the trajectories delivered by the algorithm developed in this work and the trajectories obtained with the algorithm currently used by CMC Electronics - Esterline. Thirdly, tests to compare the route selection of the LNAV five routes algorithm were performed using only the algorithm developed. The fourth and last test was a trajectory suggested by CMC Electronics-Esterline to calculate the missed approach costs.

7.1 Flight calculations validity

The first tests were performed to verify that the trajectory calculations were correctly performed. The variables to verify were fuel consumption and flight time. These variables were chosen because these two values are the ones that define the total cost of a flight. The test procedure was to run the algorithm developed with the desired input variables for a given flight, such as initial and final points, and initial weight. Then, the flight profile obtained by the algorithm was saved. The variables from the flight profile saved were KIAS/MACH climb, MACH/KIAS descent, and seven waypoints of cruise. Those values were introduced into FLSIM and the complete flight was flown. The variables of fuel and flight time were saved from FLSIM and then compared to the ones delivered by the algorithm. The flights used for this test were five flights, three Montreal – Toronto and two Los Angeles – Minneapolis. Table 8.1 shows the comparison between results obtained with these flights.

Table 7.1 Computation fidelity between the algorithm and Flight-Sim

	Fuel burned (Kg)				Flight time (Hr)			
Flight	**FlightSim**	**Algorithm**	**Δ**	**%**	**FlightSim**	**Algorithm**	**Δ**	**%**
1	4753.25	4836.45	83.20	1.75	0.6916	0.6915	0.0001	0.02
2	4853.52	4925.72	72.21	1.49	0.7173	0.7105	0.0068	0.94
3	4716.62	4812.17	95.55	2.02	0.7066	0.711	0.0044	0.44
4	18472.17	19043.99	571.82	3.10	2.9499	2.9390	0.0109	0.37
5	17995.36	18565.94	570.58	3.17	2.9799	2.9643	0.0156	0.52

It can be seen that the errors of calculations between the algorithm and FLSIM is low. Fuel burned error can be attributed to the fact that the PDB used by the algorithm does not have exact the same model as the complete aerodynamic model given in FLSIM, and to the error induced with the interpolations. Note the fuel burned error in flights 4 and 5 is higher than fuel burned error in flights 1, 2 and 3. Flights 1 to 3 were performed for a distance of 272 nm and tests 4 and 5 were performed for a distance of 1324 nm. More interpolations are needed to calculate the flight cost in flights 4 and 5 than to calculate flights 1 to 3, thus higher errors. Other parameter that suggests that the fuel burned error come from cruise interpolations is the flight time. During cruise, the flown distance, thus the calculation of flight time, is independent of the values of the PDB. Flight time is calculated depending on the aircraft speed and the selected distance in cruise. It can be seen that flight time error is always lower than 0.94% for all flight cases.

In order to have an idea of the error of the FMS CMA-9000 or PTT, 2 different flights were performed, the first flight took place from Montreal to Toronto, and the second flight from Montreal to Winnipeg on a distance of 984 nm. Results can be found in Table 7.2.

Table 7.2 Flight error between the PTT calculations and Flight Sim

	Fuel burned (Kg)				Flight time (Hr)			
Flight	**FLSIM**	**PTT**	**Δ**	**%**	**FLSIM**	**PTT**	**Δ**	**%**
1	5125.91	5399.64	273.73	5.34	0.6994	0.669	-0.03	4.35
2	15411.8	15886.4	474.64	3.08	2.2068	2.016	-0.19	8.64

Comparing the error percentages, it can be seen that the error produced by the FMS CMA-9000 (PTT) compared against FlightSim is higher than the results obtained by the routes calculated here. This can be explained with the distance of waypoints used in cruise and the calculus consideration in the acceleration phase and in the crossover calculus.

The results exposed here were only evaluated for the L-1011 because it is the only aerodynamic model available in FLSIM at LARCASE. However, the computations are valid for any other aircraft with a PDB having a similar structure to the L-1011, such as the case of the Sukhoi Russian regional jet.

7.2 Flight optimization results

Once that the fidelity of the calculus has been demonstrated, tests are done to demonstrate that the optimal profiles obtained by the algorithm here presented are better than the profiles obtained by the PTT.

The procedure to perform the tests consisting in the choice of the initial and final coordinates of the route to be followed, in the selection of a CI, and a total weight, then the algorithm was executed and the profile, fuel burned, flight time and total cost were saved. The same initial and final coordinates, weight and CI were introduced in the PTT. Then the final costs were compared. If the profiles provided by the algorithm were less expensive that the ones delivered by the PTT, the profiles were compared to ensure that they were different. During these tests, the CI selected for the L-1011 was always of 0. This was selected because the PTT available for tests has only the option of minimum fuel, which is equivalent to select a CI of 0. Nevertheless, the PTT is able to accept CIs different than 0. The PTT used to test the Sukhoi RRJ 100 is able to accept CIs different than 0. Therefore for the Sulkhoi RRJ 100 the optimization tests were performed by varying the CI.

7.2.1 L-1011 optimisation tests

In order to compare results of the optimization of the algorithm versus results of the PTT for the L-1011, 7 flights tests were performed for a CI of 0 as mentioned before. The flights are shown in Table 7.3.

Table 7.3 Flight tests for the L-1011

Flight	Departure	Arrival	Distance (nm)
1	Montreal	Winnipeg	984
2	Edmonton	Chicago	1233
3	Los Angeles	Minneapolis	1323
4	Edmonton	Ottawa	1543
5	Edmonton	Houston	1607
6	Phoenix	Baltimore	1742
7	Montreal	Vancouver	1988

These tests were performed with the algorithm and the PTT. The results were analyzed, and compared in Table 7.4. It can be seen that for every single flight, the flight cost of the algorithm was reduced with the algorithm's parameters in every single case comparing against the PTT. It can be seen that after the first flight, the economisation percentage in terms of fuel burned remains somewhat constant. The optimisation average was 1.89%. This percentage is comparable to the 1.94% found by Gagné [18]. The difference is due to the fact that Gagné calculated all the combinations of climb with cruise making t time consuming, while the algorithm explained in this work used the ratio nm/kg to define the best climb as explained in Section 5.2. Table 8.5 describes the flight profiles delivered by the algorithm and the PTT performed for every single flight given in Table 7.1. The computation time of the algorithm compared versus Gagné [18] is evident in flight 3; the algorithm presented took 12.97 seconds to solve, while Gagné's took around 30 seconds. These computations were performed in a AMD Phenom 9600B Quad-Core Processor at 2.29GHz and 4 GB or RAM.

Table 7.4 Optimisation comparison between the PTT and the algorithm

Flight	Fuel burned (kg)		Delta(kg)	Optimization (%)	Calculus time (sec)
	Algo	PTT			
1	15424.8	15886.4	461.64	2.99	7
2	19630.5	19936	305.52	1.56	10.94
3	20115.1	20371	255.90	1.27	12.9
4	23592.4	23878	285.6	1.19	12.97
5	21963.3	22602	638.7	2.82	7
6	26087.3	26683	595.7	2.23	21.4
7	29009.3	29358.14	348.79	1.20	17.81

Table 7.5 Comparison of the profiles provided by the PTT and the algorithm

Flight	Device	Climb profile	Cruise speed	Descent speed	Initial altitude (ft)	Final altitude (ft)
1	Algo	300/0.82	0.82	0.82/240	36000	36000
	PTT	311/0.824	0.824	0.8/300	38800	38800
2	Algo	300/0.825	0.825	0.825/260	36000	38000
	PTT	300/0.824	0.824	0.8/0.824	37600	37600
3	Algo	290/0.815	0.815	0.815/260	36000	38000
	PTT	290/0.79	0.824	0.8/300	38800	38800
4	Algo	300/0.825	0.825	0.825/260	36000	38000
	PTT	300/0.824	0.824	0.8/300	39800	39800
5	Algo	290/0.82	0.82	0.82/260	40000	40000
	PTT	298/0.82	0.82	0.82/300	41000	41000
6	Algo	300/0.83	0.83	0.83/260	36000	38000
	PTT	300/0.824	0.824	0.8/300	38600	38600
7	Algo	300/0.82	0.824	0.8/300	38800	38800
	PTT	311/0.824	0.824	0.82/240	36000	38000

7.2.2 Sukhoi Russian regional jet results.

Contrary to the platform for flight tests for the L-1011, the platform to test the Sukhoi RRJ 100 allows selecting different CIs. In order to compare the results of the algorithm versus the results obtained with the PTT, 5 trajectories were tested. For every trajectory, 10 flights were performed varying the CI from 0 to 90. Table 7.6 presents the flights performed and the take-off weight used in each tests. Vancouver was chosen as the maximal destination because it is near the maximal flight distance for this aircraft.

Table 7.6 Sukhoi RRJ 100 flight tests

Flight	Departure	Arrival	Distance (nm)
1	Montreal	Winnipeg	987
2	Montreal	Dallas	1317
3	Montreal	Havana	1416
4	Montreal	Cancun	1605
5	Montreal	Vancouver	1991

The global cost with the variation of CI in flight 4 is presented in Table 7.7. In Figure 7.1, the average economisation in terms of global cost from varying the CI from 0 to 90 for all flights in Table 7.7 is presented.

Table 7.7 Flight cost for a Montreal – Cancun flight with different CI

Cost Index	Global Cost (Kg)		Delta	Economisation
	Algo	PTT		
0	5641.929	5591.84	-50.0911	-0.89%
10	7851.315	7803.75	-47.5606	-0.61%
20	10059.096	9973.01	-86.083	-0.86%
30	12214.289	12100.5	-113.765	-0.93%
40	14370.089	14345.6	-24.479	-0.17%

Table 7.7 Flight cost for a Montreal – Cancun flight
with different CI (continue)

Cost Index	Global Cost (Kg)		Delta	Economisation
	Algo	PTT		
50	16525.889	16459.5	-66.429	-0.40%
60	18681.689	18573.3	-108.389	-0.58%
70	20837.489	20687.2	-150.329	-0.72%
80	22993.289	22801	-192.279	-0.84%
90	25149.089	24914.9	-234.229	-0.93%

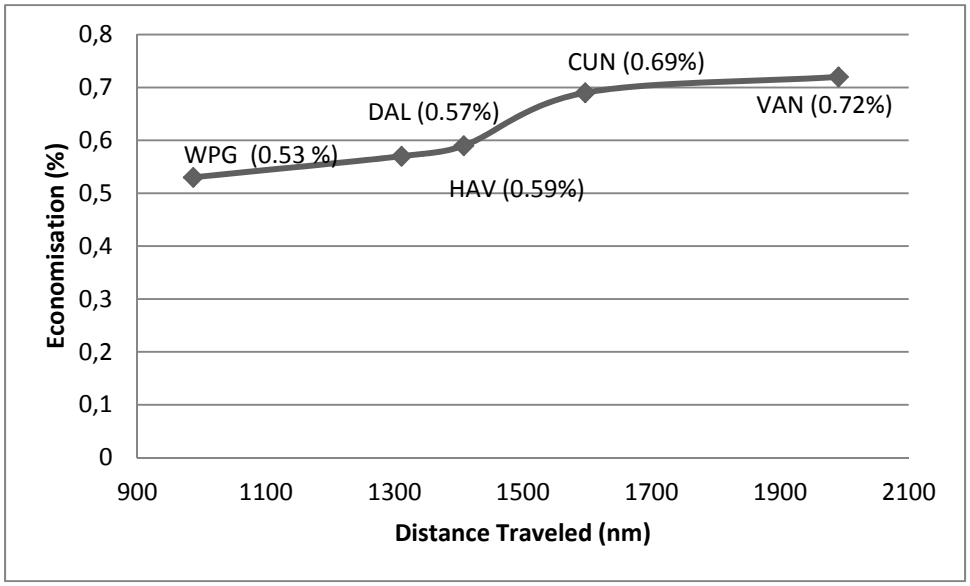

Figure 7.1 RRJ economisation for different trajectories

In Table 7.7 can be seen that for all CIs, the optimization of the algorithm gave reduced costs than the PTT. The economisation is higher as the CI increases and can be explained in the sense that the parameters selected by the algorithm allow the aircraft to fly faster thus saving flight time. However, it was noticed that the maximal cruise speed delivered by the PTT was never higher than 0.8 MACH. In the results provided by the algorithm, the fastest cruise speed delivered was 0.82 which was the maximum speed available in the PDB. Figure 7.1

shows that the optimization is higher as the flight distance increases. The optimization average for all these flights was of 0.62%.

7.3 The five routes algorithm results

The five routes algorithm is dependent on weather parameters, especially on the wind angle and speed. Testing this LNAV algorithm with FLSIM is not possible because FLSIM can accept a maximum of 16 meteorological points. The number of 16 points is not high enough to introduce the GRIB2 coordinates to evaluate the complete trajectories that the algorithm presented in this work is designed to optimise. However, the accuracy of calculations performed by the VNAV algorithm compared with FLSIM resulted precise as shown in Section 7.1. In order to test this algorithm, a given trajectory is generated and the global cost is calculated. 1 of the possible 5 routes for a given trajectory was given a more favorable tailwind to validate that the algorithm would always select this favorable route over the others. This trajectory was different than the geodesic.

The flight selected to validate the algorithm took place from Los Angeles to Minneapolis and had a length of 1,323 nm. It was selected because it was long enough to allow the wind influence to make an important difference in the flight time. The flight was performed with a weight of 175 tons. The optimal cruise for this flight is 0.81 at 36,000 ft according with the algorithm. For this test, the wind speed, angle and temperature were obtained from the GRIB2 file from Environment Canada. For the favorable route, the wind direction was fixed at an angle of 30 degrees, this is the angle that the aircraft has at the beginning of the cruise and it is the direction of the destination. The wind speed along the route is plotted in Figure 7.2. The route where the favorable tailwind is placed is the route 2 at 36,000 ft. A second flight at an altitude of 38,000 ft and MACH 0.81 was also evaluated. The same tailwind angle was imposed for the favorable route in this second flight. Note that route 1 is the geodesic route.

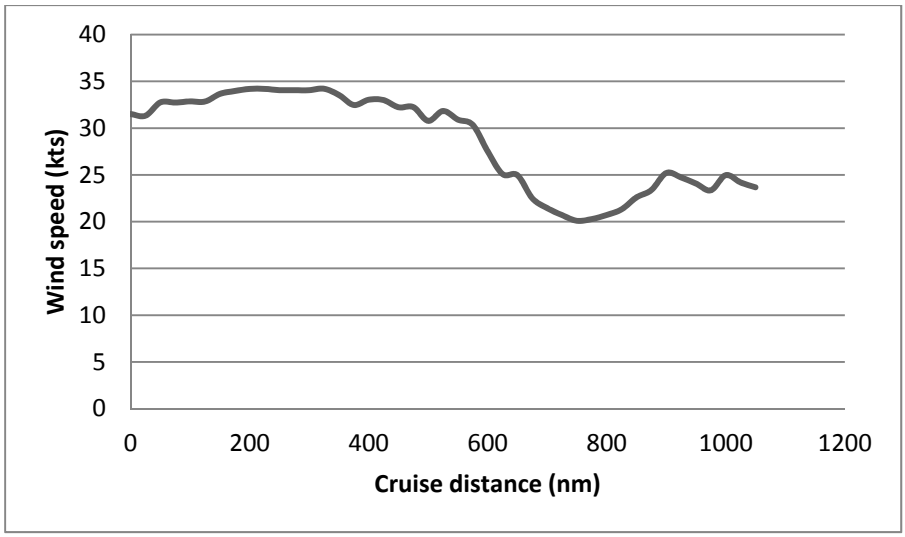

Figure 7.2 Variation of wind speed along the route

Table 7.8 Flight cost and time for the five routes algorithm

Flight		**Route 1 (geodesic)**	**Route 2**	**Route 3**	**Route 4**	**Route 5**
1	**Flight time (hr)**	3.0640	2.9007	3.0851	3.0742	3.0869
	Fuel (kg)	20516.31	19437.71	20613.74	20563.51	20651.17
2	**Flight time (hr)**	3.0872	2.9106	3.1065	3.0962	3.1087
	Fuel (kg)	20535.68	19424.05	20654.48	20603.97	20688.02

Table 7.8 shows that route 2 where the favorable tailwind was placed was selected as the optimal LNAV trajectory for flight 1 and 2. It can be seen that flight 2, which is not the optimal VNAV altitude, was always more expensive than flight 1. This was expected because the VNAV trajectory was not the optimal one.

7.4 Missed approach results

The results shown here are obtained for a flight distance of 350 nm with a climb out speed of 230 knots. Two landing approaches are displayed, the first is a successful landing procedure and the second is a missed approach procedure that was executed followed by a successful landing in the second attempt. The landing destination is the runaway 13R at the Boeing field at Seattle.

Table 7.9 Difference in consumption/emissions between full flight with a successful approach and with a missed approach

	Traveled Distance (nm)	**Fuel (lbs)**	**NOx (lb)**	**HC (lb)**	**CO (lb)**
Successful Landing	372.15	5423.95	60.96	0.70	12.71
Missed Approach	463.9	7574.02	91.11	0.93	17.54
Difference	91.74	2150.06	30.15	0.228	4.82

Table 7.10 Percentage comparison in consumption/emissions between full flight with a successful approach and with a missed approach

	Traveled Distance (%)	**Fuel (%)**	**NOx (%)**	**HC (%)**	**CO (%)**
Comparison	19.77	28.39	33.10	23.87	27.52

Table 7.11 Fuel consumption emissions between a successful approach and a missed approach

Mode	**Phase**	**Time (min)**	**Distance (nm)**	**Fuel (lb)**	**NOx(lb)**	**HC(lb)**	**CO(lb)**
LTO	**Landing**	10.86	41.4	882.31	7.4267	0.643	2.997
	Climb Out	3.49	16.07	787.59	13.495	0.037	0.708
	TOGA	0.7	n/a	132.65	2.455	0.0048	0.119
	TOTAL LTO	15.05	n/a	1802.55	23.37	0.684	3.824
CCD	**ALL**	n/a	56.42	835.61	11.6141	0.163	2.6269
TOTAL	**LTO + CCD**	n/a	n/a	2638.17	34.99	0.269	6.451

Table 7.11 Fuel consumption emissions between a successful approach and a missed approach (Continue)

Mode	Phase	Time (min)	Distance (nm)	Fuel (lb)	NOx(lb)	HC(lb)	CO(lb)
LTO	Landing	4.77	15.9	388.15	3.2672	0.0283	1.318
	Climb Out	0	0	0	0	0	0
	TOGA	0	n/a	0	0	0	0
CCD	ALL	4.77	6.253	99.95	1.5647	0.0188	0.3052
TOTAL	LTO + CCD	n/a	n/a	488.1	4.83	0.0471	1.6232
LTO	Climb Out	2.2	8.433	496.067	8.49	0.0233	0.4460
	TOGA	0.7	n/a	189.513	3.507	0.0068	0.1702
	TOTAL LTO	2.9	n/a	685.58	11.99	0.0301	0.6162
CCD	ALL	n/a	341.56	4250.25	44.116	0.6309	10.4744
TOTAL	LTO + CCD	n/a	n/a	4935.83	56.106	0.661	11.09

Table 7.12 Consumption/emissions comparison between a successful approach and a missed approach followed by a successful landing

Phase	Fuel (lb)	NOx(lb)	HC(lb)	CO(lb)
Successful Landing	488.1	4.83	0.0471	1.6232
Missed Approach	2638.17	34.99	0.269	6.451
Difference	2150.07	30.16	0.2219	4.8278
Ratio (Missed Approach / Successful Landing)	5.4	7.24	5.71	3.97

Please note that in Table 7.11, some values are not available (n/a), because it was impossible to calculate them. For example, in the LTO mode, in Table 7.11 (row 5, column 4), the speeds for the TOGA during the flight time of 0.7 seconds are not known, therefore it was impossible to calculate the distance in nm. In the CCD mode of Table 8.10, the flight time was not calculated (row 7, column 3) because the cruise speed is unknown.

As shown in Tables 7.9 and 7.10, we can observe that we have to travel close to 91.74 nautical miles (Table 7.9, column 2, row 3) more in a missed approach than in the case of a

successful landing without a missed approach landing, which translates into 28.39% more fuel burnt (Table 7.10, column 3) of the complete flight. In Table 7.11 the costs added by the take off, the cruise and the beginning of descend before the landing approach are excluded. This results in a comparison of only a successful approach and a missed approach followed by a successful landing. It can be seen that the quantity of fuel burnt is higher (Table 7.11, row 5, column 2) when the missed approach procedure is executed. Due to this extra fuel consumption an increment in all the emissions generated by this procedure is observed. (Table 7.11, row 5). Finally Table 7.12 shows the different of descents without considering the cruise. It shows that the cost of missing an approach was 5.4 times more than landing successfully.

It has been observed that the missed approach cost is highly dependent on the vector provided by ATC to get out of the holding pattern and to intercept the landing approach after the missed approach procedure; therefore attention must be focused on the ATC procedures to help ATC provide airplane return vectors that are the shortest possible, and to avoid requesting an airplane to gain even more altitude than necessary.

This difference can be explained in terms of the existence of a takeoff phase and a climb out phase. Both of these phases consume a great quantity of fuel, and thus create much more emissions. We also have to add a cruise phase at the low altitude, which is given by the ATC in the form of a vector, from which we retry the landing after the missed approach. Similar types of results are obtained for the NOx, HC and CO parameters.

CONCLUSION

The algorithm described in this thesis was successfully implemented in two different aircrafts: The L-1011 and the Sukhoi RRJ 100. The L-1011 is meant to travel long distances while the Sukhoi RRJ 100 travels medium distances. The flight distances traveled by these aircraft represent the flights distance aimed in this thesis.

The methodology followed during this work was to first develop a complete numerical model of the aircraft from the PDB including the climb, cruise and descent phases. This model calculated the cost of a given flight in terms of fuel consumption and flight time. The pre-optimal cruise algorithm was then developed to find the optimal cruise in VNAV. The "optimal" climb and the "optimal" descent were calculated using the solution provided by the pre-optimal cruise algorithm. Following, the algorithm of five routes was developed to find the optimal trajectory taking advantage of favorable winds in the route. The winds and the meteorological information were downloaded from Environment Canada and converted to MATLAB readable files. The next part of the work was to couple both algorithms, which was successfully done by performing the VNAV first, and using the optimal cruise altitude and the cruise speed, the 5 routes algorithm was implemented using this information. Finally a new method of calculating the fuel cost and emissions generated by a missed approach was developed using a database from E*MEP/CORINAIR*. Using this database, the stages of flight were divided in CCD and LTO, and calculations were performed in every stage to obtain the cost of a landing sequence and the missed approach.

Every part of the work presented in this thesis finished with a stage of validation. This validation allowed continuing working with full confidence that the preceding part of the work were results that I could rely on. This confidence helped the debugging process when errors happened to appear by only focusing on the coding being developed without worrying about the previous stages of the program.

All the objectives proposed for this work were all satisfactorily reached. For the VNAV optimization, following a comparison of the profiles delivered by the algorithm versus the profiles delivered by the PTT, when these profiles were flown in Flight-sim it was seen that the algorithm provided less expensive profiles for all cases evaluated. For the L-1011, the average optimisation over the PTT was of 1.89%, and for the Sukhoi RRJ 100, the average optimisation over the PTT was of 0.62%. The CI influence was observed during the RRJ tests. The higher the CI, the more expensive the flight was and the faster the aircraft flown as it was expected in order to save flight time.

Although improving the precision of the computation of a trajectory was not in the objectives of this work, the way the algorithm calculates a given trajectory is more precise than the way the PTT calculates the same trajectory as the results of the comparison between the PTT and the algorithm developed here when the same trajectory was tested and compared with the results provided by Flight-sim.

As for the time reduction objective, the "pre-optimal cruise" algorithm provides the optimal solution 50% faster than the algorithm developed by Gagné in [18], which was used as reference, for a trajectory of around 2,000 nm. Calculating the meteorological values by the use of the closest point method was 30% faster than using the bilinear interpolation. These time reductions satisfied this objective and make this algorithm realistic to implement in a FMS.

The objective of optimizing the LNAV was accomplished by developing the five routes algorithm. This algorithm using the wind and the temperature along the trajectory of the aircraft selects the route these meteorological variables are more favorable to reduce the fuel consumption and/or the flight time. However, because of the lack of a platform to test meteorological data is not available in this moment the testing for experimenting data was not possible, nevertheless, when favorable meteorological conditions were manually assigned to a given route, the five routes algorithm always selected that route as the optimal as it was expected. The coupling of this algorithm with the VNAV was performed by calculating the

optimal cruise altitude and speed for the first. With these speed and altitude known, the 5 routes algorithm was executed for that altitude and speed and the most economical one was found.

The last objective was to develop a way to calculate the cost in function of fuel consumption and emissions released to the atmosphere when the missed approach procedure was followed. This method was developed using data from the European monitoring and evaluation program for the aircraft Boeing 737-400. The results comparing the missed approach followed by a successful landing versus a successful landing only were close to the results suggested by Boeing. It was clear that missing an approach is expensive in terms of time and fuel for an airline; and that the extra emissions released to the environment are high such as 4.83 lbs of CO for one particular flight (to specify one emission). In this work, the execution of the missed approach procedure costs 5.4 times more fuel than carrying out the successful approach as shown in Table 8.12. It was also observed that the influence of ATC providing return vectors after the missed approach to retake the landing sequence is an important factor to work on.

The value of the research for this work can be seen as the development of an algorithm which reduced computation time; because of this reason, it can be implemented in a FMS to calculate the optimal VNAV trajectory. MATLAB coding is sometimes heavy, but the algorithm can be also be coded in different languages such as C, where the running time normally is reduced.

Another important contribution is the coupling of the VNAV and LNAV trajectory optimization. Little information is found in the literature on how to couple the LNAV and the VNAV algorithm and the work presented in this thesis introduces a way of doing this. The five routes algorithm has also served as inspiration for another method for computing the "optimal" LNAV trajectory with genetic algorithms by Roberto Felix at LARCASE.

The missed approach calculation plays an important research contribution because of the lack of information on how to calculate the cost of this procedure. The method developed here was used by CMC Electronics – Esterline to justify the development of a function to deal with this procedure in the FMS.

The last contribution of this work is the “pre-optimal cruise” algorithm to obtain the “optimal” VNAV trajectory. In this work a reduced exhaustive search was performed with the speeds and altitudes derived for the “pre-optimal” cruise profile. But this solution offers a good start to metaheuristic algorithms such as the artificial bee colony, the ant colony, the swarm particle optimization, etcetera. The “pre-optimal” cruise profile may help these algorithms to converge to the optimal solution faster than random initial values.

All this contribution were implemented in aircrafts such as the Lockheed L-1011 and the Sukhoi RRJ 100, but they can be implemented to any other airplanes with databases such as the one described here.

As it can be seen all the objectives obtained and the contributions developed in this work aims to calculate and reduce the fuel consumption, thus the emission to the atmosphere. This work helps the effort of the aeronautical industry in reducing green effect emissions to the atmosphere.

This work also may help researcher to new research topics, some of these topics are described next. The future work suggested for the “pre-optimal cruise” algorithm is to analyse it performance to determine how often it finds the optimal solution. Another good thing to work for with this algorithm is to adapt the algorithm to include the initial climb effects during the pre-cruise optimal candidate selection. In this work only the effect of cruise is evaluated, but if the climb cost is taken into account, the optimal candidate could provide good results for short flights. Although computation time is kept low, new optimisation techniques should be applied to the algorithm to reduce even more the computation time. From the meteorological data, different and friendlier databases different

than the GRIB2 files should be searched and implemented. In the present work, converting the data from GRIB2 to a file readable to Matlab takes an important amount of time and for this reason, it is impossible to implement it directly in the FMS. New techniques for the LNAV coupling should be implemented to reduce calculation time. Also in the LNAV, five fixed routes are the only options available for the algorithm. Techniques that allow the aircraft to move between these 5 routes to search for better routes can be implemented such as genetic algorithms or an improved Dijsktra's algorithm.

The algorithm presented here does not work in real time; this feature can be implemented to recalculate the optimal VNAV as the fuel consumption is being revised with the actual fuel available in the airplane. A final future work suggestion is the implementation of 4D trajectory optimization. The cost of arriving too early or late to a destination causes extra costs for an airline; impose time restrictions to waypoints or to arrive at the final destination and costs if these restrictions are not met.

For the missed approach method, the climb, cruise and descent costs are calculated for the multiple parameters such as fuel, CO, HC, etc. that are only given as one phase which is only a function of the horizontal distance traveled. In reality, the cruise phase is the only phase that is dependent on the horizontal traveled distance, while the climb and descent costs are not dependent on the flight distance, but on the targeted altitude. Therefore, the way in which the calculations are done will have to be changed to differentiate between the costs of these three phases. A different solution would consist in taking flight test data and statistically determine a percentage of the amount of fuel spent for each flight phase. In this way, the CCD data from the EIC tables could be separated for the three different CCD phases. These calculations would only be appropriate for the fuel consumption, however; a deeper analysis of the emission graphs must be done to determine if, by evaluating the fuel spent, the values of the emissions could be found in each phase (climb, cruise, and descent). Another problem that was identified was the degree of the polynomial. If this method is to be implemented in an FMS, then the calculus time for solving or/and finding the polynomial coefficients might be too high for this application to be practical.

For the missed approach, meteorological parameters have not been taken into account, but wind can have a strong influence on a flight, especially at low altitudes. Implementing the effect of the wind will change the quantity of fuel spent by the aircraft as well as the time required for it to perform the landing procedure.

BIBLIOGRAPHY

[1] Collinson, R.P.G. 2011. *Introduction to avionics systems*. 3rd edition. United States: Springer. p. 443

[2] Air Transport Action Group. 2012. *Aviation benefits beyond borders*. Annual report Online. PDF document. http://www.aviationbenefitsbeyondborders.org/sites/default/files/pdfs/ABBB_Medium%20Res.pdf. Last accessed February 28, 2013

[3] A. R. Ravishankara, John S. Daniel, Robert W. Portmann. 2009. "*Nitrous Oxide (N2O): The Dominant Ozone-Depleting Substance Emitted in the 21st Century"*. Science. Vol. 326. No. 5949. P. 123-125.

[4] Nojoumi H., Dincer I., Nataerer G.F. 2008. *"Greenhouse gas emissions assessment of hydrogen and kerosene-fueled aircraft propulsion"*. International journal of hydrogen energy. Vol 34. Issue 3. Pages 1363-1369

[5] International Air Transport Association (IATA). 2011. "*Vision 2050*". Online. 87 pages. http://www.iata.org/pressroom/facts_figures/Documents/vision-2050.pdf. Last consulted: February 28, 2013.

[6] Freitag Willian and Shulze E. Terry. 2009. "*Blended Winglets Improve Performance*". Aeromagazine Boeing. Online. 3rd Quarter. P. 9-12. <http://www.boeing.com/commercial/aeromagazine/articles/qtr_03_09/pdfs/AERO_Q309.pdf>. Last consulted: February 28, 2013.

[7] Transport Canada. 2012. "*Aircraft Fuel Reduction Initiatives – Transport Canada"*. In *Aircraft Fuel Reduction Initiatives.* Online. <http://www.tc.gc.ca/eng/programs/environment-ecofreight-air-aircraft-fuel-reduction-initiatives-96.htm>. Last consulted: February 28, 2013.

[8] Air Transport Action Group. 2009. "*Beginner's Guide to Aviation Biofuels*". Online. 24 pages. <http://www.enviro.aero/content/upload/file/beginnersguide_biofuels_webres.pdf> Last consulted: February 28, 2013.

[9] Shalom François. 2012. "*Flight path to a cleaner future*". The Gazette. Online. <http://www.montrealgazette.com/technology/story.html?id=6754763>. Last access: February 28, 2013.

[10] International Air Transport Asocciation. "*Infrastructure – Saving Fuel: 'Dive & Drive. Vs. Continous Descent Arrival (CDA)*". <http://www.iata.org/SiteCollectionDocuments/Documents/InfrastructureSavingFuelCDABrief.pdf>. Last access: February 28, 2013.

[11] Liden, Sam. 1985. "Practical Considerations in Optimal Flight Management Computations". In American Control Conference. (Boston, USA, 19-21 June 1985), p. 675-681.

[12] Liden, Sam. 1992. "*Optimum Cruise Profile in the presence of winds*". In Digital Avionics systems conference, 1992. Proceedings, IEEE/AIAA 11th. (Seattle, WA, 5-8 October 1992), p. 254-561.

[13] Liden, Sam. 1992. "*Optimum 4-D guidance for long flights*". In Digital Avionics systems conference, 1992. Proceedings, IEEE/AIAA 11th. (Seattle, WA, 5-8 October 1992), p. 262-267.

[14] Hougton Ronald C.C. 1998. "*Aircraft Fuel Savings in Jet Streams by Maximising Features of Flight Mechanics and Navigation*". Journal of Navigation. Vol. 51. Issue 3, p. 360-367.

[15] Le Merrer, Mathiew. 2010. "Optimisation de trajectoires d'avions pour la gestion du vol". ONERA Journées des Thèse 2010. (Toulouse, 25-27 January 2010).

[16] Dancila, B. D., Botez, R.M., Labour, D., 2012, "*Altitude optimization algorithm for cruise, constant speed and level flight segments"*, AIAA Guidance, Navigation and Control conference, Minneapolis, MI, US, August 13-17.

[17] Patron Felix, R. S., Botez, R. M., Labour, D., 2012, "*Vertical profile optimization for the Flight Management System CMA-9000 using the golden section search method*", IECON 2012 conference, Montreal, Canada, October 25-28.

[18] Gagné Jocelyn. 2013. "Nouvelle méthode d'optimisation du coût d'un vol par l'utilisation d'un système de gestion de vol et sa validation sur un avion LOCKHEED L-1011 TRISTAR ". Master Thesis in mechanical engineering, Montreal, École de technologie supérieure, p. 115.

[19] Fays, Julien. 2009. "Création et suivi de trajectoire en 4D avec évitement automatique des No-Fly-Zones et calculs de sorties supplémentaires pour l'aide au pilotge". Master Thesis in electrical engineering, École de technologie supérieure, p. 73.

[20] X. Prats, J. Quevedo *and* V. Puig. 2009. *"Trajectory management for aircraft noise mitigation"***,** ENRI International Workshop on ATM/CNS, Tokyo, Japan, pages 1-10.

[21] ATR. 2011. "*Fuel savings, contributing to a sustainable air transport development*". Online, p.31. < http://www.atraircraft.com/media/downloads/fuelsaving2011_1.pdf >. Last acces: 28 February 2013.

[22] Jeddi, B., J. Shortle. 2007. *"Throughput, risk, and economic optimality of runway landing operations"*. 7th USA/Europe ATM R&D Seminar, Barcelona, Spain, July 02-05, Paper 162, pages 1-10.

[23] Anderson, John D. (2005). "*Introduction to Flight"*. 6th Edition. USA: McGraw-Hill, p.392 – 394.

[24] Ojha, S.K. (1995). "*Flight Performance of Aircraft"*.1st Edition. USA: AIAA Educational series, p.249 (516p).

[25] Brandt, Steven A., Stiles, Randall J., Berti, John J., Whitford, Ray. 1997. *"Introduction to Aeronautics: A Design Prespective"*. 1st edition. USA: AIAA Educational series, p.40-46 (391p).

[26] Phillips, Warren F. 2010. *Mechanics of flight*. 2nd Edition. John Wiley & Sons, inc, p 295. (1138p)

[27] Gil, Julian. (2011). *Optimisation de la trajectoire de vol en croisière en prenant en compte l'influence du vent*. "Project Report". Montréal (Qc.): École de technologie supérieure, 62 p.

[28] Deakin, R. E., Hunter, M. N. 2009. *Geodesics on an ellipsoid – Bessel's method*. MIT University, School of Mathematical & Geospatial Sciences. Document PDF. <http://user.gs.rmit.edu.au/rod/files/publications/Geodesics%20-%20Bessel%27s%20method.pdf>. Last access : 28 February 2013.

[29] GNU Operating System. Online: <http://www.gnu.org/software/wget/>

[30] Environment Canada, Weather Office. In *Home - Analyses and Modelling*. Online: <http://www.weatheroffice.gc.ca/grib/index_e.html>. Last access: 28 February 20133

[31] National Oceanic and atmospheric administration. Online: <http://www.cpc.ncep.noaa.gov/products/wesley/wgrib2/>. Last access: 28 February 2013.

[32] Code of federal Regulations. 2010. *Aircraft Speed*. CFR Section 91.117. U.S. Online: < http://www.ecfr.gov/cgi-bin/text-idx?c=ecfr&sid=3efaad1b0a259d4e48f1150a34d1aa77&rgn=div5&view=text&node=14:2.0.1.3.10&idno=14#14:2.0.1.3.10.2.4.9>. Last access: 28 February 2013.

[33] Roberson, Bill. 2007. *Fuel conservation strategies, Cost index explained,* Aero Quarterly, 2007. Online: <http//www.boeing.com/commercial/aeromagazine/articles/qtr_2_07/AERO_Q207_article5.pdf>, pages 26-28. Last access: 28 February 2013.

[34] Namowitz, Dan. 2006. *10 Reasons to go around*, The Best of Flight Training Magazine, March 2006, Aircraft Owners and Pilots Association AOPA. Online: <http://flighttraining.aopa.org/cfis/resources/10%20Reasons%20to%20Go%20Around.pdf>, pages 1-4. Last access: 28 February 2013.

[35] The European monitoring and evaluation programme (EMEP). 2002. E*MEP/CORINAIR Emissions Inventory guidebook.* 3rd Edition. Online: <http://reports.eea.eu.int/EMEPCORINAIR3/en/B851vs2.4.pdf>, 25 p. Last access: 28 February 2013.

[36] Skyvector Aeronautical Charts . 2011. *AL-384(FAA) RNAV (RNP) Z RWY 13R Seattle/Boeing Field/ King Countr Intl (BFI)*, Online: <http://skyvector.com/files/tpp/1211/pdf/00384RRZ13R.PDF>, 1 p. Last access: 28 February 2013