HAL open science

Computation of Optimal Profiles in Descent and Approach Phases

Ramon Andreu Altava

▶ **To cite this version:**

Ramon Andreu Altava. Computation of Optimal Profiles in Descent and Approach Phases. Mathematical Software [cs.MS]. Université Paul Sabatier - Toulouse III, 2020. English. NNT : 2020TOU30026 . tel-03004543

HAL Id: tel-03004543

https://theses.hal.science/tel-03004543v1

Submitted on 13 Nov 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par l'Université Toulouse 3 - Paul Sabatier

Présentée et soutenue par

Ramon ANDREU ALTAVA

Le 17 février 2020

Calcul du Profil Optimal d'un Aéronef dans les Phases de Descente et d'Approche

---

Ecole doctorale : **AA - Aéronautique, Astronautique**

Spécialité :

Unité de recherche :

Thèse dirigée par

Daniel DELAHAYE et Thierry MIQUEL

Jury

M. Damian RIVAS, Rapporteur
M. Eric FERON, Rapporteur
Mme Itoh ERI, Examinatrice
M. John HANSMAN, Examinateur
M. Panagiotis TSIOTRAS, Examinateur
M. Daniel DELAHAYE, Directeur de thèse
M. Thierry MIQUEL, Co-directeur de thèse

Contents

List of Acronyms **xiv**

Résumé **xxiii**

Abstract **xxv**

1 Introduction **1**

1.1 General Introduction 2

1.2 Thesis contributions 4

1.3 Outline 5

2 State of the Art **6**

2.1 Introduction to Flight Management System 7

2.1.1 FMS role in the cockpit 7

2.1.2 Flight Deck Evolutions on Airbus Aircraft 8

2.2 FMS Vertical Profile Computation 10

2.2.1 Vertical path construction 10

2.2.2 Approach Path 11
2.2.3 Step-Down versus Continuous Descent Operations 11
2.2.4 Theoretical Descent Path 12
2.2.5 FMS Predictions and Guidance on the Vertical Plane 13
2.3 Energy Management 15
2.3.1 Definition 15
2.3.2 High and Low Energy Management in Approach 16
2.4 From FMS Theory to Standard Operating Procedures 17
2.5 Optimal Control Theory 20
2.5.1 Numerical Methods Applied to Optimal Control Problems 22
2.6 Trajectory Optimization Literature Review 24
2.6.1 Research Approach 27
2.7 Basics of the A* algorithm 28
2.7.1 Variants of the A* algorithm 29
2.8 Conclusion 30

3 Mathematical Model 31

3.1 Aircraft Equations of Motion 32
3.1.1 Time-dependent equations of motion 32
3.1.2 Distance-dependent equations of motion 33
3.2 Performance Models 36
3.2.1 Navigation Database (NDB) 36
3.2.2 Performance Database (PDB) 37
3.2.3 Operational Constraints 38
3.3 Airbus FMS Performance Model 39
3.4 Performance Computational Model Validation 40
3.5 BADA Performance Model 43

3.6 Optimal Control Theory 45

3.7 Conclusion 46

4 Optimal Arrival Trajectories with A* Algorithm 47

4.1 The A* algorithm 48

4.2 Implementation of A* to trajectory optimization 48

4.3 Search space generation 50

4.4 Search space pruning 51

4.5 Node generation process 52

4.5.1 Design of the approach phase 55

4.6 Data structures of open and closed lists 57

4.7 Control variables discretization 58

4.7.1 Energy share factor (ESF) 58

4.7.2 Airbrakes extension 60

4.7.3 Total flight path angle (γ_T) 61

4.7.4 Flap configuration changes 61

4.7.5 Discretization error quantification and sensitivity analysis 61

4.8 The heuristic function 63

4.8.1 Properties of the heuristic 63

4.8.2 Definition of the heuristic for trajectory optimization problems . . . 64

4.8.3 Manhattan-distance based heuristic 66

4.8.4 Flight Performance Heuristic 67

4.9 Conclusion 69

5 Results and discussions 71

5.1 Test parameters 72

5.2 Case study I: Dallas-Fort Worth arrival 73

5.2.1 Trajectory computation using BADA model 75

5.3 Case study II: Los Angeles (KLAX) arrival 80

5.3.1 Selection of the arrival procedure 80

5.3.2 Trajectory comparison with a certified FMS 82

5.3.3 Assessment of the trajectory in A320 flight simulator 86

5.4 Case study III: Aircraft high-energy condition 89

5.4.1 Aircraft high-energy condition in approach 89

5.4.2 Flight strategy comparison 90

5.5 Discussion of the results 93

6 Energy-Limit Trajectories 96

6.1 The energy-limit trajectory concept 97

6.1.1 Purpose of the function 97

6.1.2 Mathematical formulation 98

6.2 Case study at KLAX airport 101

6.2.1 Initial parameters of the simulation 101

6.2.2 Operational Assessment in the flight simulator 104

6.2.3 Visualization of the energy-limit trajectory 107

6.2.4 Conclusion 110

7 Conclusion 112

7.1 Achievements 113

7.2 Enablers for this type of flight operation 113

7.3 Perspectives and future work 114

7.3.1 Flight operations perspective 114

7.3.2 Model improvements 115

7.3.3 Heuristics using Artificial Neural Networks 116

7.3.4 From the conception towards the industrial application 116

7.4 Wrap-up 117

APPENDICES **118**

A Appendix A: Reference frames and equations of motion **118**

A.1 Reference frames and equations of motion 119

B Appendix B: Atmospheric Model **122**

B.1 The atmospheric model 123

C Appendix C: Gas Emissions Model **125**

C.1 Gas emission model 126

Bibliography **128**

List of Figures

2.1 Airbus Cockpit evolution. Source: Airbus Photolib repository. 9

2.2 Traditional Step-Down Operations (dashed line) compared with Continuous Descent Operations (CDO) (solid line) 11

2.3 Typical Flight Management System (FMS) Vertical Profile 13

2.4 FMS sub-mode request depending on altitude error 15

2.5 Reservoir analogy applied to aircraft Energy Management [1]. 15

2.6 Typical definition of stabilization point. 17

2.7 Vertical deviation on Primary Flight Display (PFD) 19

2.8 Optimal Control Problems classification 21

3.1 Representation of total flight path angle (γ_T). 35

3.2 Glide Slope beam limitation on final approach. 37

3.3 Models contained in a Performance Database (PDB) 39

3.4 Altitude (upper) and speed (bottom) profile. 41

3.5 Profile comparison between PSIMU and A* developed model. 42

3.6 Fuel and time comparison between PSIMU and A* developed model. . . . 42

4.1 A* progressive search space generation. 50

4.2 The search space at completion of the algorithm run. 51

4.3 Node generation in the presence of constraints. 52

4.4 Neighborhood zone defined around an already existing node (gray-stripped node), which is at the center of the ellipsoid. The next node n_{i+1}, generated from the current node n_i, falls in the zone. 53

4.5 Issues associated to the neighborhood area in the vicinity of a constraint. . 54

4.6 Several strategies for flight path construction on final approach. 55

4.7 Several strategies for flight path construction on final approach: u_1 idle thrust on glide, u_2 constant speed on glide and u_3 constant speed idle thrust. 56

4.8 Impact of 10% energy share increments on altitude and speed for $ESF > 0.1$. The speed target is 250 knots; small values of ESF lead to long distances while higher values yield shorter distances. 59

4.9 Impact of 10% energy share increments on altitude and speed for $ESF < 0$. The speed target is 300 knots; small negative values of ESF lead to long distances while higher values yield shorter distances. 59

4.10 Airbrakes and flap control lever on Airbus aircraft. 60

4.11 Impact of discretization in node generation process. Comparison of A* and corrected A* calculations to Runge-Kutta integration. 62

4.12 Triangle inequality property of consistent heuristics. 64

4.13 Comparison of three type of segments: geometric (dashed line), level-off plus idle (solid line) and level-off plus full airbrakes extension (dotted-dashed line). 65

4.14 Fuel consumption and time flight performance comparison. 65

4.15 Comparison between Manhattan (solid line) and euclidean distance (dash-dotted line) 66

4.16 Flight performance heuristic using Base of Aircraft Data (BADA): computation of altitude and speed profiles. For each node of the graph, the heuristic estimation is the cost from the current node to the target node. Linear interpolation to obtain this value. 67

4.17 Fuel evolution of flight performance heuristic using BADA. 68

4.18 Monotononically decreasing heuristic function. 68

4.19 Admissibility criterion of the heuristic function. 69

5.1 APPR ILS-18R. Source: NavBlue. 74

5.2 BOOVE4 Standard Terminal Arrival Route (STAR). Source: NavBlue. . . 75

5.3 KDFW case study: Altitude and speed profile. 76

5.4 KDFW case study: Fuel consumption and flight time. 76

5.5 KDFW case study: Control variables. 77

5.6 KDFW case study: Lateral path. 78

5.7 KDFW case study: Vertical visualization. 78

5.8 Flight plan preparation for KDFW arrival procedure. 79

5.9 Gas emissions comparison for KDFW case. 79

5.10 APPR ILS-24L. Source: NavBlue. 81

5.11 SEAVU2 STAR. Source: NavBlue. 82

5.12 Altitude and speed profile comparison between the reference profile computed by the FMS and the optimal one computed by the A* algorithm. . . 83

5.13 Control variables values comparison between the reference profile computed by the FMS and the optimal one computed by the A* algorithm. 83

5.14 KLAX case study: Lateral path visualization. 84

5.15 KLAX case study: Vertical path visualization. 84

5.16 FMS and A* gas emissions comparison for KLAX case study. Note that, for confidentiality reasons, the computed values can not be disclosed. . . . 85

5.17 Flight plan preparation for KLAX arrival procedure. 86

5.18 Auto-flight and thrust levers setting for the simulation tests. 87

5.19 Altitude profile comparison between the computed trajectory and the actual one flown in the flight simulator. 87

5.20 Speed profile comparison between the computed trajectory and the actual one flown in the flight simulator simulator. 88

5.21 Flight strategy 1: Altitude and speed profiles. The aircraft is too fast and too high. Flap configuration extensions are marked with numbers on the altitude profile. Priority given to deceleration. 90

5.22 Flight strategy 1: Flight control values. The extension of airbrakes is minimized while flap extensions are anticipated. 91

5.23 Flight strategy 2: Altitude and speed profiles. The aircraft is too fast and too high. Flap configuration extensions are marked with numbers on the altitude profile. Priority given to descent. 92

5.24 Flight strategy 2: Flight controls values. Airbrakes extended while required and flap extensions are delayed. 92

6.1 Non-stabilized approach as a result of strong tailwinds. 98

6.2 Influence of wind in the calculation of the trajectory. 102

6.3 Altitude and speed profile for the energy-limit trajectory. 102

6.4 Control variables for the energy-limit trajectory. 103

6.5 Energy-limit flight controls sequence. A, B, C and D represent the sequence of actions to change from clean to landing configuration. 104

6.6 EFIS and MCDU display during the simulation tests. 105

6.7 Energy-limit strategy sequence. 106

6.8 Operational use of the energy-limit trajectory. Here, the minimum distance computed by the energy-limit trajectory is displayed as the radius of the circle centered at the stabilization gate. 107

6.9 Visualization of the energy-limit trajectory through the energy-limit arc. . 108

6.10 Energy-arc (red-dashed arc) seen from the top. For a given aircraft altitude and speed, a stabilized approach is possible as long as the aircraft is beyond the arc. 109

6.11 Energy-arc in a 3-D environment. The altitude profile shown in blue represents the last trajectory that stabilizes the aircraft. As soon as the aircraft traverses the energy-arc, the approach is non-stabilized. 109

7.1 Schema of an artificial neural network. 116

A.1 Forces applied on aircraft during descent. 119

A.2 Influence of wind in flight path. 119

List of Tables

2.1 Aircraft state at the stabilization point. 16

4.1 Comparison of operation costs for typical data structures. 57

5.1 General simulation parameters for the case studies. 72

5.2 Case study I: Aircraft initial and final state. 73

5.3 List of constraints with waypoint labels for KDFW case. 73

5.4 KLAX case study initial and final conditions. 80

5.5 List of constraints with waypoint labels for KLAX case. 80

5.6 Initial and final states used for the computation of the high-energy case study. 89

6.1 Parameters of the simulation. 101

List of Acronyms

ACARS	Aircraft Communication Addressing and Reporting system
ADS	Automatic Dependent Surveillance
ADS-B	Automatic Dependent Surveillance-Broadcast
AGL	Above Ground Level
AI	Artificial Intelligence
AIP	Aeronautical Information Publication
AIRAC	Aeronautical Information Regulation And Control
ALT	A* Landmark Triangle inequality
AOC	Airline Operational Control
AP	Approach Profile
APPR	Approach Route
ARINC	Aeronautical Radio Incorporated
ATC	Air Traffic Control
ATCO	Air Traffic Controller
ATM	Air Traffic Management
BADA	Base of Aircraft Data
CAT	Commercial Air Transportation
CDA	Continuous Descent Approach
CDO	Continuous Descent Operations
CFIT	Controlled Flight Into Terrain

CI	Cost Index
CTA	Controlled Time of Arrival
DP	Dynamic Programming
EFB	Electronic Flight Bag
EFIS	Electronic Flight Instrument System
EI	Emission Indices
EPP	Extended Projected Profile
ESF	Energy Share Factor
FAF	Final Approach Fix
FAS	Final Approach Segment
FCA	Final Capture Altitude
FCU	Flight Control Unit
FG	Flight Guidance
FL	Flight Level
FM	Flight Management
FMA	Flight Mode Annunciator
FMS	Flight Management System
FPA	Flight Path Angle
GNSS	Global Navigation Satellite System
GPP	Geometric Path Point
GPS	Global Positioning System
HMI	Human-Machine Interface
IAF	Initial Approach Fix
IAS	Initial Approach Segment
ICAO	International Civil Aviation Organization
IDA*	Iterative-Deepening A*
IF	Intermediate Fix
ILS	Instrument Landing System
IMC	Instrument Meteorological Conditions
IRS	Inertial Reference System
IS	Intermediate Approach Segment
ISA	International Standard Atmosphere
JPS	Jump Point Search
LCD	Liquid Crystal Display
LCD	Reference Business Trajectory

LPA*	Lifelong Planning A*
MCDU	Multi-Purpose Control Display Unit
MCTS	Monte Carlo Tree Search
MFD	Multi-Function Display
ND	Navigation Display
NDB	Navigation Database
NextGEN	Next Generation
NLP	Non-Linear Programming
OCP	Optimal Control Problem
PDB	Performance Database
PF	Pilot Flying
PFD	Primary Flight Display
PNF	Pilot Non-Flying
RTA*	Real-Time A*
SARAH	Stabilized Approach Recovery with Automatic Handling
SDP	Soft Dynamic Programming
SESAR	Single European Sky ATM Research
SOP	Standard Operating Procedures
STAR	Standard Terminal Arrival Route
SWIM	System-Wide Information Management
TAT	Total Air Temperature
TBO	Trajectory Based Operations
TDP	Theoretical Descent Path
TECS	Total Energy Control System
TEMO	Time and Energy Management Operations
ToD	Top of Descent
TP	Trajectory Predictors
TSP	Thrust Setting Parameter
UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicle
VD	Vertical Display
VMC	Visual Meteorological Conditions

Nomenclature

$\beta_{T,<}$	Altitude gradient	[°K/m]
δ	Pressure ratio	[-]
ΔISA	Temperature difference	[°K]
δ_{ab}	Airbrakes position	[-]
Δ_{LG}	Landing gear extension	[-]
$\dot{E}$	Energy rate	$[kg/s]$
$\dot{E}_{k_s}$	Specific kinetic energy rate	$[m/s]$
$\dot{E}_{p_s}$	Specific potential energy rate	$[m/s]$
$\dot{E}_{T_s}$	Specific total energy rate	$[m/s]$
$\dot{V}$	Aircraft acceleration	$[m/s^2]$
γ	Aerodynamic flight path angle	[rad]
γ_T	Total flight path angle	[rad]
κ	Adiabatic index of air	[-]
μ	Thrust specific fuel flow (BADA)	$[kg/(min \cdot kN)]$
ϕ	Bank angle	[rad]
ρ	Air density	$[kg/m^3]$
θ	Temperature ratio	[-]
a	Speed of sound	$[m/s]$
C_D	Drag coefficient	[-]
C_L	Lift coefficient	[-]
CI	Cost Index	$[kg/min]$
D	Aerodynamic drag force	$[N]$
ds	Distance span	$[m/s^2]$

E_k	Specific kinetic energy	$[m]$
E_p	Specific potential energy	$[m]$
E_T	Specific total energy	$[m]$
ESF	Energy Share Factor	[-]
FF	Fuel flow	$[kg/s]$
g_0	Gravitational acceleration	$[m/s^2]$
h	Barometric altitude	$[m]$
h_{CSTR}	Altitude constraint	[ft]
M	Mach number	[-]
m	Aircraft mass	$[kg]$
p	Pressure	[Pa]
p_0	Standard pressure	[Pa]
p_{tropo}	Pressure at troposphere	[Pa]
R	Real gas constant for air	$[m^2/(K \cdot s^2]$
S	Wing surface area	$[m^2]$
T	Temperature	[°K]
t	Flight time	$[s]$
T_0	Standard temperature	[°K]
Thr	Thrust force	$[N]$
V	True airspeed	$[m/s]$
V_w	Wind airspeed	$[m/s]$
V_{APP}	Approach Speed	$[kt]$
V_{CAS}	Calibrated airspeed	$[kt]$
V_{CSTR}	Speed constraint	[kt]
V_{FE}	Maximum flap extended speed	$[kt]$
V_{LS}	Lowest selectable speed	$[kt]$
V_{MO}	Maximum operating speed	$[kt]$
Conf	Flap setting	[-]

In memory of my father who inspired me to pursue a career in aviation.
Your are the star that has been guiding me all along.

Acknowledgments

First of all, I would like to thank the main contributors to this PhD, who are no other than my PhD academic supervisors Daniel and Thierry and industrial supervisors Jean-Claude and Pierre. Thank you very much for your precious time, the quality of the discussions and your constructive feedback. You have certainly inspired, enriched and improved the quality of this work.

Then, I would like to thank the Airbus management team, Gilles, Nicolas and Michael for all your valuable support, which has made me grow professionally and personally during these last years. Many thanks to all members of the navigation board who showed interest in the project. In particular, I would like to thank Laurent A for your devoted dedication and constructive feedback in preparing symposiums and external communications. In the same line, I would like to thank all FMS experts and systems leaders for your time, meaningful discussions and all suggestions targeting to improve the value of this PhD. Here, I am thinking about Sylvain R, Brigitte, Julien R and Thomas P.

It has already been four years since I arrived to the FMS team, four years of professional and personal growth thanks to the people who composed the team. My special thanks go to Esteban for his quality time and for sharing his coding skills. Definitely, I have learned a lot from your ideas and your contribution has made me progress on the topic. People make places interesting beyond the intrinsic technical value. Therefore, I would like to highlight memories from the times of the FMS core team: Pierre, Xavier M, Arnaud, Thomas K, Pau, Julien R, Damien, Nicolas C, PX, Beby, Lucille, Sylvain R, Brigitte, Elodie, Gaetan and Jyoti. Thank you for the created atmosphere, I have definitively learned something from each of you. People moved, although the good atmosphere and continuous learning

has been maintained thanks to people like Guillaume, Jorge, Sylvain P, Fabien G, Didier, Anne Marie, Xavier C, Sophie, Marine, Julie, Rahul, Suraj... just to mention a few. Thank you all. Also, thanks to the transversal support provided by the assistant team composed by Véronique and Françoise. I hope that this inspiring atmosphere of the FMS family is maintained, so others can experience the same I had experienced during the last years. It has been four years of amazing technical experiences but, above all, four years of continuous learning and exchange thanks to all of you.

I would like to also thanks to all the Optim and machine learning team at ENAC for all interesting conversations. Beyond that, you have helped me to grow and reduce my 10K running average time. In the same line of though, I thank all my PhD colleagues with whom I have exchanged lots of ideas. Also, I would like to thank you the full Airlines Sciences plateau for hosting me along this project and the interesting discussions around algorithm and aviation, special thanks to Bertrand and Clement. Thanks to my previous colleagues from SII, in particular to Laureline for your wise advices and to Nicolas C for your interest in the topic. Finally, thanks to the Gazelec Navigation football team, a prove that success is not measured by prizes but by the time spent together.

Last but not least, thanks to my family and friends for all your support and patience. In particular, thanks to my parents, the values and inspiration that you transmit have helped me to become what I am today. Thank you very much with all my heart. Finally, to my love Elena, who has always been supportive with me. You have been my solid support in all moments and big part of the success is thanks to you. Thanks for being always there.

List of Publications

Publications

- R. Andreu Altava, J.C. Mere, D. Delahaye, T. Miquel and P. Neri. "Preliminary Assessment of Fuel-Efficient Permanent Arrival Profiles". Journal of Guidance, Control and Dynamics (submitted). December 2019.
- R. Andreu Altava, J.C. Mere, D. Delahaye and T. Miquel. "Graph-Search Descent and Approach Trajectory Optimization Based on Enhanced Aircraft Energy Management". AIAA Aviation Forum. Dallas, United States, 17-21 June 2019.
- R. Andreu Altava, J.C. Mere, D. Delahaye and T. Miquel. "Flight Management System Pathfinding Algorithm for Automatic Vertical Trajectory Generation". 37th Digital Avionics System Conference. London, United Kingdom. 24-28 Sept 2018.

Patents

- R. Andreu Altava and J.C. Mere. "Distance minimale pour dissipation de l'énergie lors d'un atterrissage". Airbus Operations SAS. (Filed dossier). Reference: 16088. Filing date: 25-11-2019.
- R. Andreu Altava and J.C. Mere."Procédé et dispositif pour générer en temps réel une trajectoire verticale optimale destinée à être suivie par un aéronef". Airbus Operations SAS. Filing number: 1873084. Filing date: 17-12-2018.

Résumé

Le contexte actuel de croissance du trafic aérien, qui double tous les quinze ans, pose des problèmes environnementaux et remet en cause le développement durable de l'aviation. De plus, d'autres facteurs comme l'entrée en vigueur de nouveaux décrets relatifs aux questions environnementales, la volatilité des cours du pétrole et aussi la concurrence exacerbée du marché des compagnies aériennes conduisent au fait que les sujets de recherche liés à l'optimisation fine du profil de vol de l'avion et à l'amélioration de l'efficacité des opérations aériennes sont devenus des enjeux majeurs pour l'aviation.

Le système de gestion du vol, ou FMS selon l'acronyme anglais, est un système de navigation embarqué, courant dans tous les avions de transport commercial, qui permet à l'équipage de gérer le plan de vol latéral et vertical. Du fait que les systèmes avioniques aient des performances limitées, les algorithmes embarqués font des calculs sur la base d'hypothèses très conservatrices. Ceci conduit à des écarts notoires entre les calculs du FMS et le profil réellement volé par l'avion dans un environnement dynamique du vol. L'objectif de cette thèse est donc de développer une fonction bord intégrée au concept de poste de pilotage des futurs cockpit Airbus, permettant de générer des trajectoires optimisées mais aussi tenant compte de l'environnement dynamique de l'avion. Pour cela, cette nouvelle fonction bord qui a été développée adapte la stratégie et le profil de vol de façon régulière pour minimiser le coût global de l'opération.

Les principes de gestion énergétique d'un aéronef sont utilisés pour optimiser le profil vertical de vol dans les phases de descente et d'approche dans le but de réduire la consommation carburant, les émissions de gaz à effet de serre et potentiellement le bruit généré par les moteurs et les surfaces aérodynamiques. La fonction proposée est basée sur les

principes de la programmation dynamique et plus particulièrement sur l'algorithme A*. Elle cherche à minimiser une fonction de coût en traversant un espace de recherche généré au fur et à mesure que l'algorithme avance dans ses calculs. Non seulement la trajectoire résultante est optimale mais aussi relie la position courante de l'avion avec le seuil de piste de l'aéroport d'arrivé indépendamment du mode de guidage et des conditions énergétiques, ce qui est une nouveauté par rapport au FMS. Les résultats sur simulation montrent que la consommation carburant est réduite de 13% et que les émissions de gaz à effet de serre de 12%. De plus, l'algorithme propose une stratégie de vol pour dissiper l'excès énergétique dans le cas de sur-énergie, où l'avion est trop haut en altitude et/ou trop rapide en vitesse. La représentativité opérationnelle des profils calculés a été évaluée dans les simulateurs de vol Airbus. Ces tests sur simulateur démontrent que les profils calculés peuvent être suivis par l'équipage avec les modes de guidage existants, même si une automatisation serait souhaitable vis-à-vis de la charge de travail. Enfin, cette thèse constitue une base solide pour la génération en temps réel de profils optimisés de descente et d'approche afin d'automatiser l'exécution de ces phases de vol.

Abstract

The continued increase of air traffic, which doubles every 15 years, produces large economic benefits but poses environmental issues that put at risk the sustainable development of air transport. Other factors such as jet fuel prices volatility, the introduction of new environmental regulations and intense competition in the airline industry, have stimulated in the last years research on trajectory optimization and flight efficiency topics.

The Flight Management System (FMS) is an onboard avionic system, standard in all transport aircraft, which is used by flight crews to manage the lateral and vertical flight-plan. Since current avionic systems are limited in terms of computational capacity, the computations performed by their algorithms are usually done on the basis of conservative hypotheses. Thus, notorious deviations may occur between FMS computations and the actual flight profile flown by the aircraft. The goal of this thesis is to develop an onboard function, which could be integrated in future Airbus cockpits, that computes optimal trajectories, readjusts the flight strategy according to the dynamic aircraft condition and minimizes operating costs.

Flight energy management principles has been used for optimizing aircraft trajectories in descent and approach phases with respect to fuel consumption, greenhouse gas and noise emissions. The proposed function has been developed on the basis of dynamic programming techniques, in particular the A* algorithm. The algorithm minimizes a certain objective function by generating incrementally the search space. The exploration of the search space gives the optimal profile that links the aircraft current position to the runway threshold, independently of the current flight mode and aircraft energy condition. Results show 13% fuel savings and a decrease of 12% in gas emissions compared with a

best-in-class FMS. Furthermore, the algorithm proposes the flight strategy to dissipate the excess of energy in situations where aircraft fly too high and/or too fast close to the destination runway. A preliminary operational evaluation of the computed trajectories has been conducted in the flight simulators. These tests demonstrate that the computed trajectories can be tracked with current guidance modes, although new modes should be required to decrease the workload of flight crews. In conclusion, this paper constitutes a solid background for the generation of real-time optimal trajectories in light of the automation of descent and approach flight phases.

1

Introduction

This chapter describes the current commercial aviation context and introduces trajectory optimization topic as a manner of improving flight efficiency. The main contributions and the outline of this thesis are also exposed at the end of the chapter.

1.1 General Introduction

Air traffic has experienced a continuous growth in the last decades and is expected to be doubled by 2030 [2]. According to Airbus global market forecasts [3], 33.000 aircraft will enter into service in the next 20 years, doubling current global aircraft fleet. Eurocontrol forecasts four possible scenarios depending on the geopolitical situation in the 2040 time horizon concluding that, in the most likely scenario, traffic in Europe will be 1.5 times that of 2017 [4]. As a consequence, airspaces volumes will get increasingly congested, specially in continental areas, generating delays in the network. Besides, it is expected that Unmanned Aerial System (UAS) gradually integrate the same airspace. Jet fuel prices have presented high volatility in the last decades and are expected to increase in the short term. Airlines tough market competition makes them to seek for solutions that optimize their business routes, open new ones and propose associated services to increase their market share. Environmental issues are a big concern for the aviation community that aims to reduce noise and gas emission levels in order to make aviation sustainable in the long term [5]. In this context, a modernization of the Commercial Air Transportation (CAT) at all levels seems paramount to cope with the previously mentioned challenges. Major projects as Single European Sky ATM Research (SESAR) in Europe [6] and Next Generation (NextGEN) [7] in the United States foster research activities that provide operational solutions to modernize current Air Traffic Management (ATM) system. On one hand, NextGEN focuses on the use of Global Navigation Satellite System (GNSS) and augmentation systems in order to improve flexibility of airports operations and reduce the dependency on ground infrastructure and data communications whilst providing with new automation tools to improve en-route and terminal area operations. On the other hand, SESAR initiatives follows a similar roadmap whose main contribution aims at improving operation efficiency, providing a more flexible and optimal use of the airspace. In general terms, safety levels are expected to be improved, with a raising concern in cybersecurity and more specifically in data securing topics [8]. In parallel, CleanSky programme [9], launched by the European Commission, aims to foster environmentally friendly flight operations in order to enhance nowadays noise abatement procedures. The implementation of a System-Wide Information Management (SWIM) [10] platform to efficiently share flow information among all stakeholders is an important contributor to Trajectory Based Operations (TBO)[11]. In that context, airlines have a higher degree of freedom to plan the most suitable 4D Reference Business Trajectory (LCD), while being compliant with Controlled Time of Arrival (CTA) at several stages of flight. Similarly, Extended Projected Profile (EPP) [12] aims to down-link FMS aircraft intended trajectory to ATC centers so that ground situation awareness is improved and airlines have better chances to fly their intended routes [13]. Thus, an efficient air traffic system is of common

interest to airlines, manufacturers, authorities and service providers in order to face current and future challenges [14].

The natural tendency in commercial aviation community has been to automate those prone-error tasks, sometimes repetitive and tedious (with higher probability of making mistakes), and the role of pilots smoothly transitions from piloting to monitoring tasks. Technology is currently available to support this tendency. On one hand, on-going studies on Artificial Intelligence (AI) [15] and machine-learning techniques may construct the bridge towards the new generation of aircraft, which relies on automation as its first mean. It is a novel paradigm as, nowadays, avionics logics are defined based on a series of scenarios and it is responsibility of the pilot to cope with the situation when an event out of this logic appears. On the other hand, ground infrastructures and communications with airborne means have evolved. For instance, on-boarded Automatic Dependent Surveillance-Broadcast (ADS-B) equipment is prepared to send parts of the flight-plan to the Air Traffic Control (ATC), a basic enabler for 4D operations. It seems a logic evolution that proven technologies in unmanned vehicles are implemented in certified aircraft for commercial purposes in the coming years. The challenge here is to implement these technologies to support the growing demand while safety levels are maintained, if not improved.

Trajectory optimization has been a topic of interest for years in the research community, most of the projects with a focus on flight planning, weather avoidance and trajectory prediction for ATM. However, new challenges highlight the need to go a step further and propose novel paradigms and operations that adapt, on a real time basis, flight strategy to the surrounding environment so that better predictability and savings could be achieved. From an airlines perspective, flight planning is an essential part of their business as it permits to minimize operation costs. Current flight operations are full of uncertain events, the most common being aircraft deviations from their routes imposed by Air Traffic Controller (ATCO) and weather conditions during flight. In order to face uncertainty, airlines start to include data from past flights in their flight planning algorithms, thereby they can anticipate those events in future flights. This approach works quite well for flight planning purposes and could be combined with deterministic on-boarded automation to enhance flight operations efficiency.

In this thesis, aircraft energy management principles are applied to optimize aircraft descent and approach paths, which can potentially generate large savings in terms of fuel consumption, delays, noise and gas emissions whilst reducing the number of non-stabilized approach and go-around events. The concept may be seen as an extension of present CDO, which can be compatible with nowadays aviation framework. Additionally, these trajectories are computed accounting for the dynamic environment in which the aircraft evolves.

These trajectories define the optimal flight strategy to be followed by pilots in order to manage aircraft energy, within reasonable boundaries, in an efficient way. Therefore, the research of this thesis aims to improve FMS design with respect to the construction of descent and approach paths, and to provide an optimal and permanent trajectory that accounts for the current aircraft position independently of its energy condition. For the purpose of clarity, the permanent trajectory is defined as the trajectory linking the runway threshold to the current aircraft position, independently of current flight modes and aircraft energy condition.

1.2 Thesis contributions

The primary contributions of this thesis are:

- A generalized algorithm that computes, for any arrival procedure containing any set of constraints, the optimal descent and approach path of a commercial aircraft. On the basis of the A* algorithm, this solution corresponds to the global optima of a certain objective function, in most cases, fuel consumption. The trajectory is optimal and permanent, which means that the calculation reaches the current aircraft position regardless of current guidance mode or aircraft energy condition.
- In approach phase, when the aircraft is close to the runway threshold and in high-energy condition, which means it flies too high and/or too fast, the algorithm provides a dynamic re-computation of flap extensions, landing gear and airbrakes deployment, as it is required to dissipate the excess of energy and perform a stabilized final approach.
- An extension of the algorithm to compute the energy-limit trajectory, which is the upstream trajectory that stabilizes the aircraft according to the current energy state in the minimum ground distance. The provision of this information to flight crews improves the energy awareness with the aim of reducing the number of non-stabilized approaches and go-around procedures.
- The analysis of the results suggests that traditional fixed speed descent paths, which are based on Mach/CAS coupling provided by pilot-entered Cost Index (CI), could be improved thanks to the variable selection of the optimal speeds. It means a transition from optimized speed profiles to optimized energy profiles, which decreases discontinuities occurrences on the vertical flight-plan.
- Relevant fuel savings, which are of the order of 13%, and reduction of greenhouse gas emissions for the analyzed case studies.

On one hand, the main scientific contribution of this thesis is the demonstration that dynamic programming, in this case implemented by means of a version of A* algorithm, is a pertinent approach to highly-constrained Optimal Control Problem (OCP)s. In practice, these constraints are used for the incremental construction of the search space, represented as a graph, and to prune those flight strategies which do not satisfy the published flight procedure. Moreover, the *curse of dimensionality* can be beaten through the use of meaningful heuristic functions, which can be defined by means of known information of the problem or state-of-the-art predictors such as neural networks. On the other hand, this thesis proposes an important contribution to the industry, since it reopens the debate about the generation of more efficient trajectories than those generated by current FMS. Besides, the methodology proposed aims to compute an optimal and permanent trajectory, whose flight strategy could be followed manually by flight crews or automatically by appropriate guidance laws. The automation of labor-intensive flight phases, such as descent and approach, may pave the way of more automated systems in the light of efficient flight operations.

1.3 Outline

The document is structured as follows: chapter 2 exposes best-in-class FMS path construction in descent and approach phases and a literature review of trajectory optimization studies, with emphasis on Optimal Control and graph search techniques. Then, chapter 3 formulates mathematically the problem and presents two aircraft performance models used by the algorithm for the calculations. Furthermore, chapter 4 describes in detail the implementation of the algorithm on the basis of the A* algorithm, which generates optimal trajectories in descent and approach phases. This chapter also includes a review of the properties of well-defined heuristic functions and proposes some heuristics for trajectory optimization problems. Chapter 5 presents three different case studies, whose results are discussed and compared with those of a state-of-the-art FMS. A preliminary operational evaluation of the computed trajectories on flight simulators is also presented. In addition, chapter 6 presents another case study for which energy-limit trajectories are computed. Finally, chapter 7 summarizes the contributions of the thesis, outlines directions for future work and suggests recommendations in light of an industrial application.

2

State of the Art

This chapter is structured in different parts. The first one introduces the role of FMS in nowadays cockpits with focus on how reference descent and approach paths are generated and tracked in the current operational framework. The second part proposes a review of optimal control methods to solve trajectory optimization problems. Then, the literature review serve as the basis for the justification of the research approach proposed in this thesis. Finally, the chapters ends with the introduction of the main principles of the A algorithm and other path-finding variants.*

2.1 Introduction to Flight Management System

2.1.1 FMS role in the cockpit

To summarize, the FMS is a complex avionic system integrated in the Auto-Flight System (ATA22), whose principal aim is to reduce pilot workload. Depending on the standard, also known as the software version, embedded in the FMS, functional content may vary. It contains all functionalities that are used to perform a basic function. The main functions [16] of the FMS are briefly described hereinafter:

- **Navigation**: In most aircraft families except A350, Flight Management (FM) computes the aircraft position based on data coming from different sources, such as Global Positioning System (GPS), Inertial Reference System (IRS) and ground-based beacons (radionavigation). Nominal operations usually rely on the combination of GPS signal with inertial data, since inertial drifts over time are corrected by the GPS itself.

- **Flight Planning**: This function is responsible for the construction of a flight plan based on pilot's entered data via Multi-Purpose Control Display Unit (MCDU) Multi-Function Display (MFD), data retrieved from Navigation Database (NDB) or data sent by the company via Aircraft Communication Addressing and Reporting system (ACARS) or datalink.

- **Lateral and Vertical Guidance**: The FM module computes and sends guidance commands to the Flight Guidance (FG) module, which sends the orders to flight controls computers with the consequent deflection of wing surfaces, which modify the attitude of the airplane.

- **Performance computation**: This function embedded in the FM partition encloses aerodynamic and engine performance models. A closed-loop computation is performed between the vertical and the lateral path defined in the flight path, since the lateral path depends on the speed profile constructed by the integration of the equations of motion. Furthermore, this module computes target speeds, vertical predictions along the flight plan and the construction of a reference vertical profile, which will be detailed in the next chapter.

- **Datalink**: It constructs, receives and interprets Airline Operational Control (AOC), ATC and Automatic Dependent Surveillance (ADS) messages conveyed via ACARS protocol through the dedicated equipment.

- **Provision of information to display systems**: The FMS conveys data via Aeronautical Radio Incorporated (ARINC) 429 protocol to the Electronic Flight Instrument System (EFIS), which processes and displays that information on PFD and Navigation Display (ND) screens. Moreover, the system acts as an Human-Machine Interface (HMI) between the pilot and aircraft systems.

The equipment was put into service in 1982 [17], when it was on-boarded in Boeing 757 and 767 aircraft. In 1983, A310 became the first Airbus[1] jet to integrate the system, becoming an avionics standard equipment for the following generations of aircraft. The introduction of FMS brought great benefits to the aviation community, specially for airlines, which experienced relevant fuel cost savings. The system decreased the workload and contributed to reduce flight crew to two members, removing the flight engineer post from the cockpit. In that flight deck configuration, the role of the pilot was no longer limited to pure piloting tasks but to manage the flight through this system. In some way, the appearance of FMS supposed a change player in aviation as pilots were not only required to focus on piloting skills but also on other avionics systems. New functional contents have been introduced since then into FMS standards, accentuating the role of the pilot as a flight manager.

Later on, FMS was able to request flight information to AOC through ACARS or, latterly introduced, datalink communication. Operationally, pilots request flight plan, performance, wind and temperature data to the dispatch centre, which is sent and confirmed by the pilot upon manual action. Current FMS contains a series of functions to check the validity of the data, triggering alerting messages in case a discrepancy is found or a safety risk is encountered. Part of the pure tasks done by pilots, as entering parameters into the FMS, were progressively delegated to the dispatch centers. The gain was twofold; pilot workload was alleviated and prone-error tasks were reduced.

2.1.2 Flight Deck Evolutions on Airbus Aircraft

Flight decks on Airbus aircraft have evolved during the last decades due to the introduction of new technologies, moving from mechanical to electronic instruments. Advancements in avionics systems have permitted to remove the flight engineer from the cockpit, as those functions were performed in the FMS with the information being displayed to the pilot when needed. Formerly steam cockpits 2.1(a) consisted of a vast quantity of analogue dials and gauges, which were substituted by glass cockpits 2.1(b), these consisting of a series of Liquid Crystal Display (LCD) screens to display the information to the pilot.

[1] At that time the contract was signed by Aérospatiale on behalf of the European consortium Airbus Industrie

Early glass cockpit, for instance in A300, combined first versions of EFIS for attitude and navigation information with traditional mechanical gauges for airspeed, variometer and altitude. Then, full-glass cockpits 2.1(c) replaced traditional instruments, whose unique use is limited to back-up instrumentation, by electronic ones. Following this evolution, A380 and A350 cockpits presented larger displays 2.1(d) than their predecessors, an unequivocal sign that monitoring tasks are increasingly relevant in nowadays pilot role. This evolution in Airbus cockpits can be seen in Fig 2.1.

(a) A300 steam cockpit.

(b) A310 early glass cockpit.

(c) A320 full glass cockpit.

(d) A350 Cockpit with large displays.

Figure 2.1: Airbus Cockpit evolution. Source: Airbus Photolib repository.

Regarding the avionic systems, best-in-class FMSs embarked in Airbus aircraft have been continuously improved over the years, adding new functional content at each standard delivery with the aim of reducing pilot workload and operating costs for airliners [18]. These novel functions have reduced the workload of flight crews, whose role is gradually transitioning from piloting to system management and monitoring. In this sense, the autopilot can be engaged several seconds after taking-off and, theoretically, flight crews can fly in managed modes until the start of the approach phase. Hence, take-off and landing are the unique flight phases in which the intervention of pilots is mandatory [8].

This thesis focuses on descent and approach flight phases for a series of reasons, which are not related with pure optimization of the flight. In this sense, studies have shown that

greater benefits in terms of fuel reduction are obtained when optimizing cruise and climb flight phases instead of descent and approach. According to [19], around 49% of fatal accidents occur during approach and landing phases. This is probably due to the fact that pilots have more tasks to manage during these flight phases, increasing workload and stress, which are contributing factors to loss of awareness events. The surrounding environment is complex as they are restricted by ATC, which provide radar vectors that bring aircraft out of their intended flight plans. In these scenarios, pilots have to manage the energy state of the aircraft in order to avoid as much as possible unstabilized approaches, which result in a workload increase. For this purpose, this thesis proposes a decision-aid tool that permits to generate a trajectory that connects the runway to the aircraft position and whose computation is based on optimal energy management, which reduces noise and fuel consumption. It decreases pilot workload and could eventually automatize descent and approach phases, reducing the likelihood of initiating a go-around.

2.2 FMS Vertical Profile Computation

2.2.1 Vertical path construction

FMS computes a reference path once the arrival procedure is entered into the system, flying it as long as managed modes are activated. For the sake of simplicity, FMS separates the lateral and vertical path construction, both referenced through aircraft ground speed and distance to destination for each waypoint. This thesis maintains this hypothesis with the consideration that the lateral path is already known and provided by an external function that performs the optimization. As for the vertical profile, FMS performs two types of computations; an off-line part where the reference profile is constructed and an on-line part where FMS predicts aircraft state all along the flight plan. FMS computes a vertical trajectory that complies with all procedure-constraints and that is optimized with respect to a certain CI defined by the airline, which can be defined as the ratio between time and fuel costs. In general, time cost is attributed to maintenance, delays, marginal depreciation, leasing costs and personnel while fuel cost is subjected to market price fluctuations and may vary significantly among geographic sectors. Vertical path is composed of a descent profile, called Theoretical Descent Path (TDP) and an Approach Profile (AP), both computed backwards from runway threshold to cruise altitude. A trajectory in the vertical plane can be defined as a combination of segments that define an altitude and speed profile. FMS construction takes into account flight envelope limitations as well as operational constraints related to instrumental flight procedures defined in the Aeronautical Information Publication (AIP), which are coded within the NDB using A424

international standard.

2.2.2 Approach Path

The AP starts at runway threshold and finishes at deceleration point, which delimits the start of deceleration towards approach speed. The AP is formed by three segments: Final Approach Segment (FAS), Intermediate Approach Segment (IS) and Initial Approach Segment (IAS). The FAS is constructed from the runway threshold to the Final Capture Altitude (FCA) or Final Approach Fix (FAF), depending on the type of procedure, while the IS goes from the FAF to the Intermediate Fix (IF). Then, the IAS is cosntructed from the IF to the Initial Approach Fix (IAF), which is the entry point to the procedure as published in the approach chart. The deceleration point is usually located in the middle of the IAS. Aerodynamic configuration points are located at specific characteristic speeds computed by the performance module. FAS construction depends on the type of approach; precision approaches usually impose a slope of approximately -3° whilst non-precision approaches are defined by altitude minima. Furthermore, the construction of the AP, specially regarding the IS and the IAS, differs between Continuous Descent Approach (CDA) and classical step-down approaches.

2.2.3 Step-Down versus Continuous Descent Operations

Conventional step-down operations are those where aircraft deceleration is done in a level-off segment usually between the range of 3000 - 5000 feet in order to increase deceleration efficiency, while in CDO profiles, aircraft decelerates along the path as it descends. The comparison of both profiles is observed in Fig. 2.3.

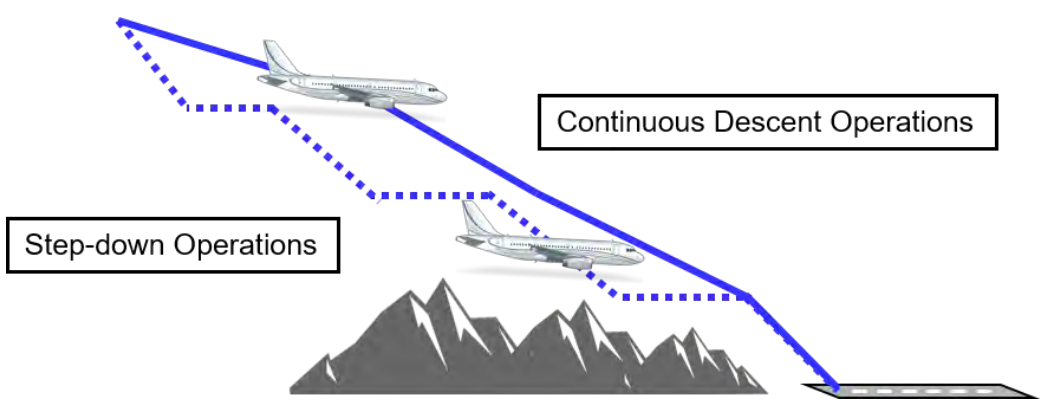


Figure 2.2: Traditional Step-Down Operations (dashed line) compared with CDO (solid line)

This is the main goal of CDA function, which is an enabler of CDO. The main advantage of this type of approach is that aircraft descent is continuous and avoids deceleration segments at low altitudes, which has an impact on noise, pollutant gas emissions and, in some cases, fuel consumption [20]. Regarding fuel consumption, large debate has been generated about the fact that CDA sometimes consume more fuel than classic approaches, since idle ratings are higher in a certain flap configuration than in clean configuration. Aircraft deceleration is more efficient in a level-off segment than in CDO, and depending on the arrival procedure could lead to long approach phases as the deceleration is started at high altitudes (e.g. 8000 feet). However, CDO reduce considerably noise emissions.

2.2.4 Theoretical Descent Path

Once the altitude and speed profiles are defined for the approach part, TDP is constructed from deceleration point until cruise altitude, where the Top of Descent (ToD) is located. This path consists of a concatenation of idle and geometric segments, the latter being those generated when an altitude constraint restricts the construction of the [idle] path. In general, geometric segments require auto-thrust adjustments to maintain a speed target while the elevator guides the aircraft through the vertical path. In contrast, idle segments set auto-thrust to idle while the elevator maintains the target speed. The transition between geometric to idle path is defined by the Geometric Path Point (GPP), which corresponds to the last encountered constrained waypoint. Geometric segments are constructed with a constant (ground) flight path angle. For doing so, a flight path comparison method is used by the FMS to define the type of segment depending on the altitude constraints and aircraft descent capability. In the nominal case, geometric segments require extra thrust to maintain the flight path. Half-airbrakes segments are constructed in case the aircraft needs extra drag to satisfy the altitude constraint. Finally, if half-airbrakes are not sufficient to satisfy that constraint, a vertical discontinuity is created in the flight path, also known as a too steep segment. The optimal descent speed (Mach/CAS) is computed by the FMS depending on the CI value entered by the pilot. This speed is used for the whole descent except when any speed constraint applies. Decelerations from optimal speed to constrained speeds are performed through a defined Energy Share Factor (ESF), a parameter that distributes the available energy (loss) between altitude and speed. The Mach/CAS optimal coupling divides the idle path into two segments; the first one (in the upstream direction) where the aircraft descends at constant optimal CAS until the crossover altitude is attained and the second, where the aircraft flies at optimal Mach speed until the cruise flight level. In order to avoid overshoot, a profile capture segment can be added before intercepting the cruise altitude, which slightly delays the ToD position.

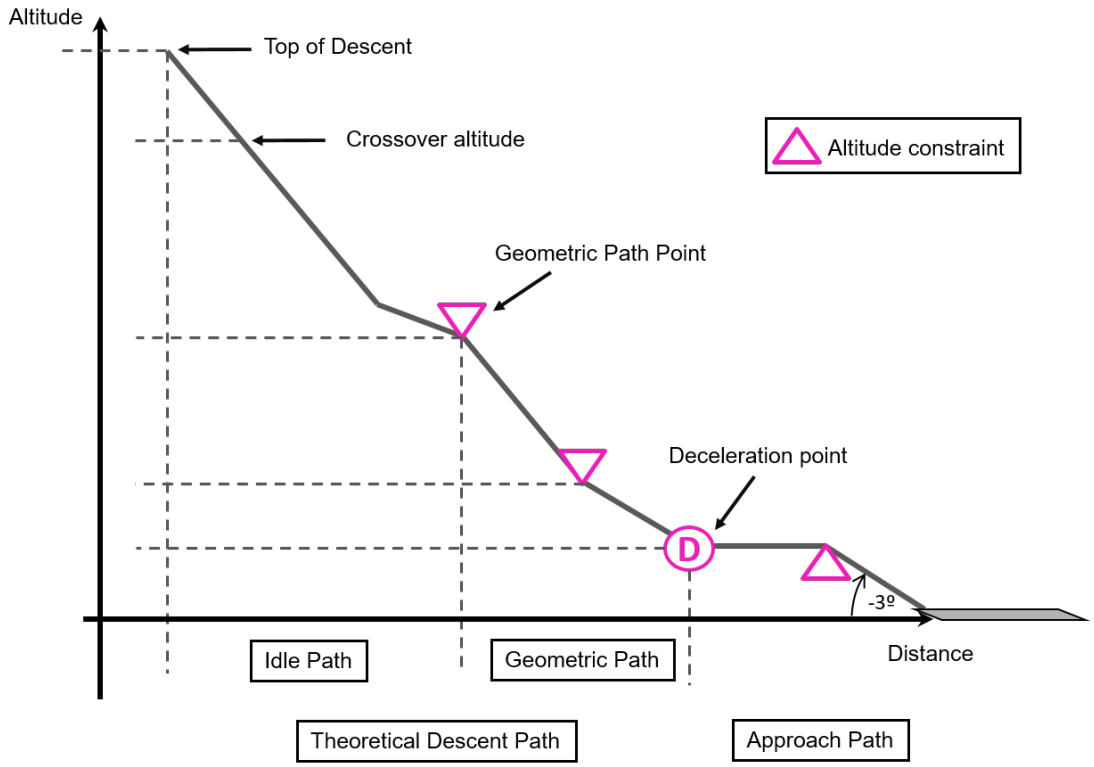


Figure 2.3: Typical FMS Vertical Profile

Current vertical profile design is optimized with respect to a selected CI in case that no biases exist between the aircraft behaviour and FMS hypotheses. FMS assumes immediate profile recapture as soon as the aircraft is off-vertical path. However, if the aircraft is deviated from its intended lateral path, the vertical path is no longer valid as the computation is based on a lateral trajectory that is not being followed by the aircraft. This is why vertical managed modes (DES mode) cannot be engaged if the aircraft is not in lateral managed mode (Nav mode). Under these circumstances, pilots disregard FMS predictions as they are probably based on hypotheses different than the current condition. The profile is recomputed only in certain conditions, for instance, when a "direct to" action is taken by the pilot (i.e. FMS computes a straight segment to a waypoint assigned by the pilot). However, these recomputed trajectories do not attain aircraft position and pilots are responsible for the proper management of aircraft state in order to reduce the altitude error, a process that may lead to inefficient energy management if the wrong flight strategy is selected.

2.2.5 FMS Predictions and Guidance on the Vertical Plane

Apart from the vertical reference profile, aircraft predicts forwardly its state along flight plan waypoints based on the same state integrators than for the vertical profile computa-

tion. Predictions are regularly updated to be consistent with aircraft condition but based on hypotheses that may be different than the aircraft actual behavior. These predictions provide to the pilot a good idea of aircraft intentions but shall be only interpreted as advisory information. In scenarios where aircraft is off-lateral path, predictions become less useful since they are based on hypothesis that may differ from the aircraft actual behavior over time.

Regarding the guidance part in descent and approach flight phases, FMS sends pitch and thrust targets to the guidance module, which is in charge of guiding the aircraft through the reference trajectory generated by the FM. However, due to uncertainties such as non-accounted degraded engine performances, unforeseen strong winds and other operational constraints imposed by ATC, aircraft may deviate from that vertical path. In that case, FMS requests a sub-mode engagement to the FG module with the assumption that the profile is recaptured as soon as possible, which depends on the altitude and speed errors with respect to the reference profile. In summary, three logics apply as resumed in Fig. 2.4:

(i) **On path**: In this scenario the aircraft tends to follow the vertical reference profile (TDP), FMS requests VPATH/SPD sub-mode where the altitude path is followed by the elevator and the speed target is maintained by the auto-thrust. In low altitudes, priority is given to the path rather to the speed. Under these circumstances, aircraft keeps the flight path and, in case of an unexpected tailwind, kinetic energy increases instead of deviating from the profile, which is then compensated by a pilot action.

(ii) **Below path**: When the aircraft is placed below the reference profile for any reason (usually early descent or head wind gust), the FMS requests VS/SPD sub-mode where the aircraft rejoins the profile with a fixed vertical speed target, while the auto-thrust adjust thrust to maintain the target speed.

(iii) **Above path**: In this situation, the aircraft tries to rejoin the vertical profile as soon as possible. For doing so, SPD/THR sub-mode is requested by the FMS, where the elevator controls the speed (speed target plus a delta for steeper descent) and the engines are set to idle. Interception of the reference profile is displayed on ND, through a pseudo-waypoint that assumes half-airbrakes extension for a quicker interception.

These modes are only engaged when lateral NAV [managed] mode is engaged. Pilots monitor FMS predictions as they give an idea whether a constraint is going to be missed or satisfied. Additional pilot actions such as airbrakes extension in descent, and flap configuration changes or landing gear deployment in approach may be required in case those sub-modes logics are not sufficient.

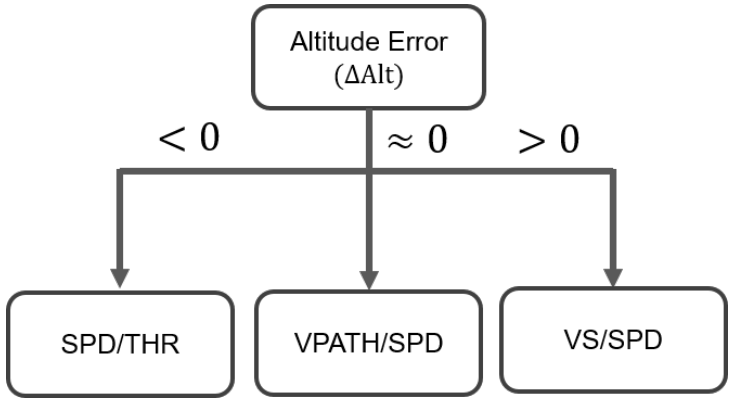


Figure 2.4: FMS sub-mode request depending on altitude error

2.3 Energy Management

2.3.1 Definition

In flight operations, the term energy management refers to the continuous exchange that occurs between potential and kinetic energy. According to the law of conservation of energy in physics, energy is neither created nor destroyed but transformed from one form into another. The level of energy of an aircraft is defined through basic parameters such as airspeed, airspeed trend, altitude, drag and thrust [21].

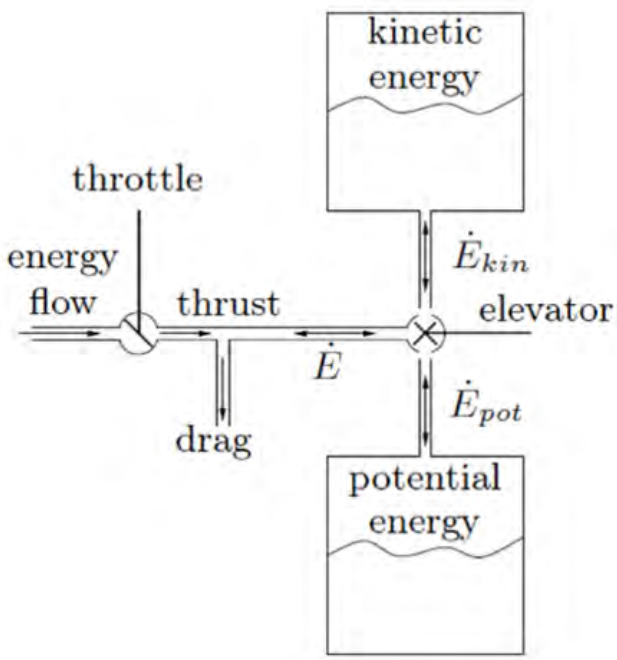


Figure 2.5: Reservoir analogy applied to aircraft Energy Management [1].

Considering the total energy of the aircraft as the sum of kinetic (E_k) and potential energy (E_p), chemical energy from fuel can be transformed into mechanical through thrust whereas aerodynamic drag produces a continuous exchange between mechanical and heat energy, which impacts on the aircraft total energy state. The reservoir analogy proposed by [1] is a good representation of the energy exchanges that take place in an airplane: as observed in Fig. 2.5, throttle levers and drag devices such as airbrakes, flaps and landing gear control the flow of energy, which is then distributed between kinetic and potential reservoirs through the elevator.

2.3.2 High and Low Energy Management in Approach

Pilots ensure that the energy level is appropriate with regards to the flight phase, correcting it if necessary. High-energy occurs when the aircraft is too fast, too high or both, while in low-energy situations the aircraft is below its vertical path or target speed [22]. These situations frequently occur due to wind errors or ATC instructions bringing the aircraft off-path in order to manage the surrounding traffic and may require pilot intervention. In managed modes, different guidance modes are engaged to correct deviations based on aircraft position with respect to the reference profile. On one hand, when aircraft is above path, auto-thrust sets thrust to idle while the elevator keeps a speed target. This mode, called SPD/THR, allows to perform a steep descent and using airbrakes contributes to reduce the vertical deviation. On the other hand, a low vertical speed is maintained by the elevator while auto-thrust keeps the target speed. This mode, called VS/SPD, needs additional thrust to keep the flight path and target speed is usually lowered to be close to the optimal glide speed, defined in Airbus lexicon as greendot, which maximizes the lift-to-drag ratio. In approach phase, good energy management prevents pilots from aborting landing. Pilots have several strategies to correct the energy level of the aircraft and the outcome depends on their skills.

	Aircraft state
Distance, NM	-2.96
Altitude, ft	1000 AGL
Speed, kt	V_{APP}
Confaero	Full + Landing gear

Table 2.1: Aircraft state at the stabilization point.

An approach is said to be stabilized if the aircraft is on glide slope flying at final approach speed in landing configuration at a height of 1000 ft Above Ground Level (AGL)

in Instrument Meteorological Conditions (IMC) or 500 ft in Visual Meteorological Conditions (VMC) [23]. This is usually called the stabilization point and is defined as per the parameters defined in table 2.1 and the construction given in Fig. 2.6. Energy mismanagement have a direct impact on flight operation efficiency and sometimes on safety, since unstabilized approaches require to initiate a go-around procedure [24].

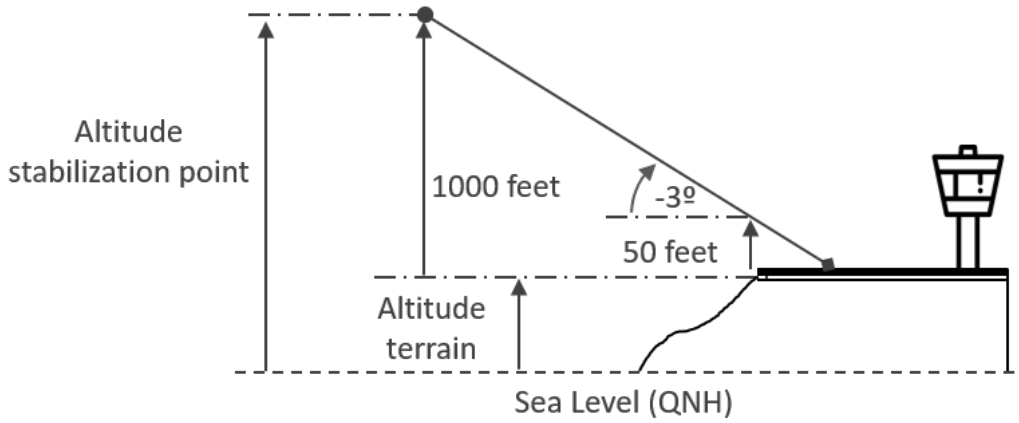


Figure 2.6: Typical definition of stabilization point.

Low-energy states are solved by correcting altitude or speed with extra thrust. In high-energy situtations, it is more complex as pilots have the choice of deploying airbrakes, anticipating flaps or extending landing gear to overcome the event. In addition, some of these decisions are irreversible, since flaps or landing gear cannot be retracted except in go-around procedures. That brings pilots to a situation where they have to estimate, based on their experience, the best policy that limits the impact on flight efficiency. Airlines are interested in decision aid and automated tools that propose pilots with safe and optimal trajectories, since pilot workload is reduced focusing their attention in tasks such as monitoring and communication.

2.4 From FMS Theory to Standard Operating Procedures

Current Standard Operating Procedures (SOP) rely on FMS to provide position and an effective flight planning to pilots, whose information is displayed either on the EFIS (CDS for A380/A350), consisting of two LCD screens; ND and PFD; or on the MCDU (MFD for A380/A350). MCDU is used for long-term management while EFIS displays information for piloting purposes. The HMI of the system with the pilot is done via the MCDU/MFD.

In an ideal scenario, pilots follow FMS reference profile either manually or automatically with the Auto-pilot (AP) and Auto-Thrust engaged [25]. In most operations, spe-

cially in congested airports, pilots do not usually fly in managed modes but in selected or manually. According to data analysis performed by Airbus customer support teams [21], 99% of aircraft tune a speed target. This technique is usually applied by ATCOs to ensure aircraft separation and assign time gains or delays, as necessary. Radar vectoring is another technique highly applied in dense airports as traffic flows are easily managed and predictability is improved. These radar vectors modify aircraft lateral route with the consequence operational impact for the pilot. Imposed vertical speeds are seldom used by ATCOs but by pilots to monitor easily the rate of descent of the aircraft with respect to their mental calculations. Any ATCO instruction has a direct impact on the optimality of the flown trajectory as it restricts aircraft motion. Nowadays, a common ATC practice is to clear an aircraft for a certain airspace volume, usually bounded by two altitude constraints. In the worst case, aircraft speed is imposed but it can still optimize the altitude profile. Apart from the above exposed reasons, aircraft leave their vertical trajectory for a set of typical reasons listed hereinafter:

- **Clearance Altitude:** ATC may not clear an aircraft to descend, i.e. the pilot is not allowed to dial down Flight Control Unit (FCU) altitude to proceed with the descent. Instead, a level-off segment is maintained by the aircraft until ATC provides clearance for a lower altitude. In this scenario, the profile is not recomputed when the ATC clearance is obtained and the aircraft will be usually above the profile.

- **Holding pattern:** Similar to the previous scenario, ATC may request one aircraft to perform a holding pattern or pilots may request ATC to perform it for some reason (e.g. not prepared for landing). In this case, the aircraft may continue its descent if clearance was already obtained. In most cases, the aircraft will be located above the profile as well.

- **Wind error:** It stands for a wind that was not forecast and consequently was not entered into FMS wind page for profile construction. Usually, flight plans are computed several hours before departure so that winds may change in this period of time. Wind gusts or turbulences induce aircraft speed to increase or decrease depending on the wind direction (headwind, tailwind, crosswind or vertical gusts). This speed difference with respect to the theoretical descent speed, even if temporary, puts the aircraft above or below the TDP.

- **Too Steep Path:**This type of segment constructed by the FMS occurs when an altitude constraint cannot be satisfied even with half airbrakes extension hypothesis. The heavier the aircraft, the less descent capability. It results in a discontinuity in the vertical path, which locates the aircraft above path as soon as the constraint is overflown.

- **Other causes:** This category can include simplistic models, not accounted performance degradation due to aircraft aging (engine performance, fuel flow estimation, aerodynamics). Basically, inappropriate fleet management and monitoring may result in non-accounted performance degradation. This causes a discrepancy between FMS models and actual aircraft performance, leading to inaccurate computations. To avoid it, most airlines perform one-to-one optimization for each aircraft of their fleet, which reduces aircraft exploitation costs.

Altitude error is displayed on PFD and MCDU through the vertical deviation label. Figure 2.7 displays vertical deviation "yoyo" concept, a green circle displayed on the altitude scale that shows the deviation from the TDP based on the current aircraft altitude. Vertical deviation accurate value is displayed on FMS Progress page. Pilots are aware of aircraft energy state and estimate the best flight strategy to perform a stabilized approach, acting on flight control surfaces consequently. Thus, they are fully responsible for managing the total energy of the aircraft. This study proposes a decision-aid tool that computes the optimal flight strategy for a stabilized approach taking into account the current aircraft energy state. The advantages of this approach are numerous, since it decreases non-stabilized approaches and go-around procedures, flight crews workload during approach at the same time as some criterion is optimized.

Figure 2.7: Vertical deviation on PFD

2.5 Optimal Control Theory

Before entering in the details of OCP, a differentiation between trajectory optimization and OCP [26] is introduced. The first term, frequently known as motion planning in robotics, is a kind of problem where inputs to the dynamic system are parameters, whose optimum values yield to an optimized trajectory. Nonetheless, the inputs in OCP are not parameters but functions, commonly dependent of time. In that case, the objective is to find the optimal control input that complies with an optimization criterion, being more appropriate to use the term OCP.

Optimal control theory is an extension of the calculus of variations [27], which was largely popularized in the 1950's, mainly after the contributions of Richard Bellman and Lev Pontryagin. The arrival of the digital era was a big enabler for the application of this theory on several fields as aerospace, robotics, economics, among others. Before that, only simple problems could be solved and its application was very limited. The idea behind this kind of mathematical optimization is to find the best values for state, $x(t) \in \mathbb{R}^{\mathrm{n}}$, and control variables, $u(t) \in \mathbb{R}^{m}$, that minimize or maximize a cost function:

$$J = \Phi\Big[x(t_0), t_0, x(t_f), t_f, p\Big] + \int_{t_0}^{t_f} \mathcal{L}\Big[x(t), u(t), p, t\Big] dt \tag{2.1}$$

Where p is a vector containing a set of static parameters and t is the independent variable time. The previous Eq.(2.1) is optimized over the period of time $t \in [t_0, t_f]$, where both $t_0, t_f \in \mathbb{R}$. The objective function is subjected to a series of dynamic constraints defined by the equations of motion of the vehicle:

$$\dot{x} = f\Big[x(t), u(t), p, t\Big] \tag{2.2}$$

Also several path constraints apply to the problem:

$$g_l \leq g\Big[x(t), u(t), p, t\Big] \leq g_u \tag{2.3}$$

The subindex l indicates the lower bound whereas u refers to the upper limit. Then, the boundary conditions [28] at time t_0 are described as:

$$\phi_{0_l} \leq \phi[x(t_0), u(t_0), p, t_0] \leq \phi_{0_u} \tag{2.4}$$

Whereas the terminal conditions at time t_f:

$$\phi_{f_l} \leq \phi[x(t_f), u(t_f), p, t_f] \leq \phi_{f_u} \tag{2.5}$$

A particular case of Eq.(2.3) are bounds on state variables, $x(t)$, defined by:

$$x_l \leq x(t) \leq x_u \tag{2.6}$$

And also those applied to the control, $u(s)$:

$$u_l \leq u(t) \leq u_u \tag{2.7}$$

In other words, the goal is to find over the time t, the control inputs $u(t)$ of a dynamic system $\dot{x}$, which satisfies the physical constraints g, and optimizes a specified performance index J [29]. In most cases, OCPs are separated into different phases, $k \in [1, ..., K]$ using a linkage function. The multi-phase problem is expressed as follows:

$$J = \sum_{k=1}^{K} J^{(k)} \tag{2.8}$$

The problem can be formulated differently so that it can be solved through different methodologies. One of the most popular classifications divides OCPs in four classes, as it is displayed in Fig.2.8:

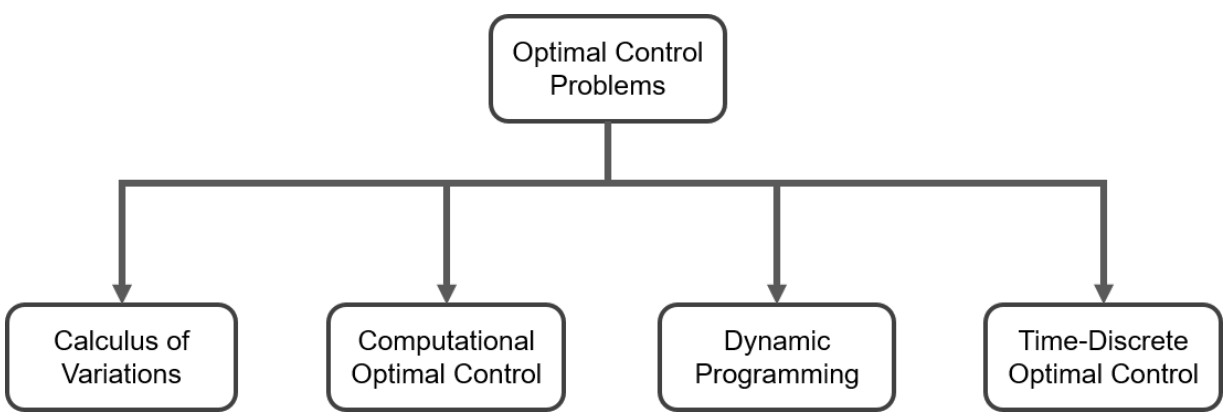


Figure 2.8: Optimal Control Problems classification

2.5.1 Numerical Methods Applied to Optimal Control Problems

The traditional approach to solve OCPs was to find analytical solutions through the application of the theory of calculus of variations. However, due to the innate complexity of these problems, limitations with constraints on state and control variables and, notably, with the development of computers, numerical methods replaced this kind of approach [30], which is highly abandoned for aerospace applications as solutions are unlikely to be found analytically. On numerical methods side, solving methodologies are classified into two major categories: indirect and direct methods [31].

On one hand, indirect methods rely on the Hamiltonian and the derivation of first-order necessary conditions to find the best solution that satisfies costate and/or interior points. The Hamiltonian $\mathcal{H}$ is defined as:

$$\mathcal{H}(x, \lambda, \mu, u, p, t) = \mathcal{L}(x, u, p, t) + \lambda^T f(x, u, p, t) + \mu^T g(x, u, p, t) \tag{2.9}$$

Where λ and μ are vectors applied to f and g functions containing different Langrange multipliers. Indirect methods apply the Pontryagin maximum (minimum) principle [32], which is developed from the calculus of variations theory:

$$\mathcal{H}(x^*(t), u^*(t), \lambda^*(t), p, t) \leq \mathcal{H}(x^*(t), u, \lambda^*(t), p, t) \tag{2.10}$$

Which can be written in the most generic form as:

$$u = arg \min_{u \in U} \mathcal{H} \tag{2.11}$$

Where U defines the set of feasible controls. In the particular case where the control is continuous, the first-order necessary conditions for optimality, commonly known as Euler-Lagrange equations, are defined as:

$$\frac{\partial \mathcal{H}}{\partial u} = 0 \tag{2.12}$$

$$\dot{\lambda} = -\frac{\partial \mathcal{H}}{\partial x} \tag{2.13}$$

The OCP is usually converted into a two or multiple-point boundary value problem [33][34]. The main drawback of this methodology is that obtaining analytical expressions may be complicated for certain complex non-linear systems (depending on the model, the derivation of the Hamiltonian becomes impracticable) and the accuracy of the solution is dependent on the initial guess. Furthermore, constraints on state and control variables

are difficult to handle, especially the latter, which may lead to discontinuities. Numerical examples of indirect methods are gradient and shooting methods [33].

On the other hand, direct methods discretize infinitely-defined continuous OCP, which is transcribed into a finite dimensional Non-Linear Programming (NLP) problem. Then, it is solved through mathematical techniques to solve NLP problems [35]. One of the most popular methods is the collocation methods [36] [37] including Hermite-Legendre-Gauss-Lobato. Another popular technique is based on pseudospectral methods [38] [39], Legendre or Chebysheb polynomials [40], whose goal is to search the coefficients of those functions as optimisation parameters, despite Gibbs phenomenon [41]. As they have become part of an intensive study, there is a large quantity of software that proposes transcription of OCP to NLP quite easily. The reason why direct methods have been thoroughly studied is related to the convergence required to find a solution. However, large-size problems may lead to large execution times, making these methods undesirable for on-boarded applications. Other source of problems is the grid definition and the application of smoothing techniques to ensure the feasibility of the solution [42][43]. Moreover, the final solution depends on the initial guess and the methodology does not ensure to obtain the global optima.

Another approach that has been increasingly studied, after the work of Richard Bellman on Dynamic Programming (DP), is based on the Bellman's Principle of Optimality [44][45]. This method relies on the use of the Hamilton-Jacobi-Bellman optimality criterion to find a solution using a recursive approach [46]. Basically, the objective of dynamic programming theory is to divide the problem into a set of sub-problems, which are linked together, then solve and store each of them them for later use (memoization technique). In DP, the dynamic system is discretized as follows:

$$x_{k+1} = f(x_k, u_k) \tag{2.14}$$

Where x_0 is known, $x_k \in \mathbb{R}^{\mathrm{n}}$, and $u_k \in \mathbb{R}^m$. The performance function is defined as follows:

$$J = \sum_{k=0}^{N-1} r(x_k, u_k) \tag{2.15}$$

The Bellman optimality principle defines the cost-to-go function R_i at the given time i, which is solved recursively.

$$R_i(x_i) = \min_{u_i}\Big(r(x_i, u_i) + R_{i+1}\big(f(x_i, u_i)\big)\Big) \tag{2.16}$$

The main disadvantage of this method is what is called as the "curse of dimensionality", that is, the number of computations increase exponentially as state and control variables

increases because more combinations have to be explored at each step.

Other less popular but interesting methods found in the literature that can be applied to OCP are evolutionary algorithms, e.g. genetic algorithms [47], and path-searching solving methods such as Dijkstra's [48] and A* [49] [50], which are special cases of DP. Nevertheless, the main difference between Dijkstra and A* algorithm is that the latter uses a heuristic function that avoids to explore the whole search space of the problem. Other researches have used hybrid approaches, which combine direct and indirect methods or mixing one of them with other optimization techniques [51][52] [53].

2.6 Trajectory Optimization Literature Review

Lots of works in the literature have tried to solve OCP by applying different methodologies and strategies. The conclusion that can be inferred from their reading is that there is no ideal methodology to solve all of them but the choice should be made based on the intrinsic problem. Comparative studies such as [54][29][26] discuss the pertinence of using one or another approach, highlighting the advantages and drawbacks of each technique. The trajectory optimization problem is commonly interpreted in two ways: a strategic approach where optimization is performed on ground [55][52](flight planning and cost index choice) and a tactical approach, whose optimization is usually done on-board on a real-time basis. Ground means rely on Trajectory Predictors (TP) in order to enhance ATC situational awareness and aircraft predictability. Most researches in ATM domain use generic aircraft performance models to predict aircraft intentions. For this purpose, [56] develops a machine learning model to detect on a real-time basis aircraft guidance modes with the aim to estimate aircraft trajectory.

For the sake of simplicity, lateral and vertical flight paths are frequently treated separately, while other works opt to construct 3D (or 4D) optimal trajectories directly [57][58]. Besides, different optimization criteria are taken, being fuel consumption the most popular criterion to minimize among them. Total Energy Control System (TECS) focuses on optimal energy management function [59] in order to implement a control law to manage aircraft total energy of the aircraft. A proportional control law substituted pitch (autopilot) and thrust (autothrust) control algorithms. Flight test campaigns on a B757 at NASA facilities were performed successfully and positive feedback was received from pilots but TECS logics showed the same level of complexity as state-of-the-art control laws. [60] proposed algorithms for on-boarded systems with the aim of optimizing en-route and terminal area segments. This approach varied conveniently the energy share factor before integrating backwards the equations of motion, which presented similarities with nowadays FMSs. Airbus already has worked in energy management concepts from a flight

controls perspective as in [61], which divides the energy management problem in short and long term functions. Then, both are combined into one function, named Stabilized Approach Recovery with Automatic Handling (SARAH). On one hand, short-term energy management is done through airbrakes and engine regime control. On the other hand, the long term prediction module estimates the optimal point where aerodynamic configuration and landing gear are extended, which ensures proper energy management to perform a stabilized approach.

Most of the recent researches apply direct methods, as it is the case of [62], which proposes a Time and Energy Management Operations (TEMO) algorithm that constructs a reference energy-neutral trajectory (i.e. no additional thrust and no speed brakes extension), based on CDO profiles, from the stabilization point to the ToD backwards, with a fixed arrival time at the FAF. In order to satisfy a controlled time of arrival constraint, the algorithm deviates on purpose from the reference profile to cope with disturbances, with the goal of minimizing airbrakes extension and thrust solicitation. The study considers both strategic and tactic computations; in the former, the profile is recalculated when boundaries are exceeded. In the latter, a tactical approach, allows slight deviations from the reference profile, which are absorbed through flight controls. The main objective of this work is to reduce fuel consumption and environmental impact, based on CDA profiles and assuming idle thrust settings all along the descent profile. The performance of that algorithm is compared with a state-of-the-art FMS in [63], whose results show that TEMO concept reduces time errors and fuel consumption when compared with a simulated FMS.

The work presented in [36] proposed an algorithm based on collocation methods that solved the mathematical transcription of the optimal control problem and suggested several refinements to model realistic problems. 4D optimazion is performed in [39] through Gauss pseudo-spectral conversion into a NLP that is then solved through a sequential quadratic programming method. GPOPS solver is used in [38] for computing optimal trajectories and assess and quantify the environmental benefits, through noise and gas emissions models, with respect to recorded data retrieved from Frankfurt airport. Indirect methods are applied in [64] [65] for real-time computation without path constraints. Computation burden is relatively low and results are compared with a state-of-the-art FMS for time and fuel optimization problems.

As for applications of the dynamic programming theory, [66] proposed an optimal control problem solved by a Soft Dynamic Programming (SDP) algorithm to calculate optimal 4D trajectories, in the presence of one or more time constraints in the flight plan. The problem is discretized and solved by means of the Bellman principle of optimality. A kind of heuristic function decreases computation time by reducing the size of the state space, through the introduction of operational limitations and aircraft performance constraints.

A two-layer neural network replaces traditional computations process and improves the total performance of the algorithm. Fuel savings are quantified at around 2% of the total fuel consumption during the cruise phase. [67] proposes an optimal control algorithm to compute 4D-optimal trajectories in the presence of wind, by reducing fuel consumption all along the flight. It uses a dynamic programming approach to solve the optimization problem. Results are then compared with real flight data obtained from an airline covering a regional trip in Japan, yielding an average of 10% savings in terms of fuel consumption but penalizing flight time by the same percentage. In the UAS domain, [68] proposes a lateral path planner algorithm, called Kinematic A*, that finds free-obstacle optimal path of a Unmanned Aerial Vehicle (UAV) taking into account dynamics, so trajectory generation and guidance is done at once without requiring any flight plan smoothing. Studies like [49] [50] implement A* algorithm in order to compute optimal descend paths for pre-defined gridded graphs. As a consequence, the quality of the solution strictly depends on factors such as state space discretization and branching factor value, and only consider typical descent procedures with simple time heuristics that lead to bad performances. Other works have also proposed a solution for the dynamic computation of a flight strategy in approach, as it is the case of the Low Noise Augmentation System (LNAS) presented in [69], which computes dynamically airbrakes, landing gear and flap settings extension in order to stabilize the aircraft whilst minimizing noise impact. Besides, the function has been embarked on an Electronic Flight Bag (EFB) and has been successfully flight tested.

To conclude, trajectory optimization is an OCP solved through a very large range of methods in the literature. In general, cruise is the preferred flight phase to optimize as it consumes most of the carried fuel while climb-to-approach optimization is the second choice for most researchers. Nevertheless, some of those approaches seem not to be consistent from an operational point of view, since they do not take into account all operational constraints but a set of constraints associated to a nominal operation [52]. Other approaches are not intended for on-board assistance as their algorithms converge into a solution in a considerably high execution time, being merely developed for flight planning purposes rather than for real-time optimization. Several studies focus on Energy management principles [70] as a mean to enhance CDO although few focus on optimizing the approach path [71]. First of all, the present work aims to go deeper into details of descent and approach operations and take into account actual constraints for any arrival procedure in the world, so the vertical path is optimized based on aircraft performance capability. In a second time, energy management principles are applied to readjust the optimal profile, on a real-time basis, according to the dynamic and uncertain operational environment.

2.6.1 Research Approach

In general, the literature agrees to model trajectory optimization problems as an optimal control problem solved by numerical methods. Reference [72] describes the vertical motion of an airplane as a system of equations involving seven variables, five of which are state and two of which are control. Per definition, state variables are those whose derivatives are present in the equations of motion. A system composed of seven variables and five equations has two degrees of freedom. To integrate them, a best-in-class FMS makes hypotheses that remove the remaining degrees of freedom and the ordinary differential equations through a classic trapezoidal-method. An optimization model can be implemented if the performance index is represented as a function of state and control variables. Though, intense debate is found in the literature regarding to the method used for solving this kind of problems.

On one hand, indirect methods were assessed in a first stage of the thesis but results were unsatisfactory as convergence was not ensured as soon as the problem became complex. The main disadvantage was that for [complex] constrained problems, the derivatives of the Hamiltonian were too complex and computation time exploded. Besides, handling constraints on control variables was a difficult task [73]. On the other hand, direct methods are a good alternative but they usually rely on external solvers for solving the NLP problem, which act as black boxes, hence being less desirable for on-board purposes (e.g. GNOPS, SNOPT, IPOPT). Indeed, direct and indirect methods, which are similar by nature except for the way the optimization is performed, are based on typical operations for which they define a set of typical segments, each of them being a phase of the multi-phase problem. This formalism makes the algorithm lose generality as those approaches are not implementable for solving any type of arrival procedure for any airport in the world. On the opposite side, dynamic programming algorithms, whose main drawback is the "curse of dimensionality", work quite well when a large set of constraints apply, since they reduce large areas of the search space. Actual flight operations typically contain a large set of constraints such as altitude and speed constraints defined by the arrival procedure, flight envelope, accelerations and time. This was the main reason behind the choice of dynamic programming for solving the optimal control problem. Out of the discussion are stochastic algorithms such as Simulated Annealing and Evolutionary algorithms due to certification issues. The particular method of A*, which applies dynamic programming to find the shortest path in a graph, is a special case of Branch and Bound algorithms. In general, it is an interesting choice as no other graph search algorithm finds the optimal trajectory in less time than A*, as long as the heuristic function is properly defined. In brief, A* is deterministic, complete, easily implementable as a portable function and improves the efficiency, in terms of time-complexity, with respect to similar graph search methods.

2.7 Basics of the A* algorithm

In computer science, A* is a best-first search algorithm that was first published in 1968 [74] and originally intended for autonomous robot motion in constrained environments. Nowadays, the algorithm has numerous applications on computer games and cartographic platforms like Google Maps or Waze. A* combines Dijkstra's and greedy best-first search algorithms and is considered as a special case of Branch&Bound algorithm belonging to the family of DP methods. This graph traversal algorithm finds the shortest path using the cost already traveled (Dijkstra's) and a estimation of the remaining cost (Greedy Search). Information about the problem is used to construct a heuristic function that guides the solution reducing time-complexity. Considering a graph defined as a set of nodes and arcs $G(n, a)$, A* algorithm assigns to each node n a score $f(n)$, which is based on the cumulative travel cost, $g(n)$, and an estimated cost, $h(n)$, until the target:

$$f(n) = g(n) + h(n) \tag{2.17}$$

In the particular case where $h(n)$ is zero, A* algorithm develops exactly the same nodes as Dijkstra's. The optimality of the solution is guaranteed if the heuristic function is admissible, i.e. it always under-estimates the optimal cost from the current to the target node. It is a tricky trade-off as the closer from the optimal cost that estimation is, the fewer nodes the algorithm expands. In addition, if the heuristic function is consistent, that is monotonically decreasing all along the path, the algorithm only needs to develop a node n once and A* algorithm is complete. Moreover, in the case the heuristic function is admissible and consistent, A* finds the global optimum and implements a list with all candidate nodes, called open list, and another with those nodes already explored (closed list). In case of developing a node that is already in the open list, only the least cost node is retained. In general graph theory, trees implement only an open list whilst graphs include as well a closed list, so trees may visit the same node several times [75] [76]. Thus, graph search requires more memory while tree search usually takes longer computation times.

Graph representation is an important part of graph theory as it has a direct impact on performance and the quality of the solution [77]. Most common representations are grids (e.g. for robot motion planning), polygonal vertex (e.g. lateral weather avoidance), road maps (e.g. commercial road network applications) or waypoints (e.g. a drone in a restricted environment). Nodes of the graph can be computed ahead of time or being progressively generated as the algorithm progresses. This choice merely depends on the application; generally, generated graphs provide with an extra of flexibility and require in

general less memory than pre-computed graphs. This work proposes an approach where nodes are gradually generated because, the only nodes known at the beginning of the computation are the current and the final aircraft state, so those in-between shall be computed taking into account aircraft performance, weather conditions and constraints.

2.7.1 Variants of the A* algorithm

It is worth to cite other relevant variants of the A* algorithm developed with the aim of mitigating problems related to time and space complexity. For instance, hierarchical path planning uses abstraction to reduce the problem into a set of simpler sub-problems, also known as layers, and then put the final solution together. Similarly, A* Landmark Triangle inequality (ALT) [78] algorithm selects a set of landmarks to compute a lower-bound to find the optimal path of the original problem. In dynamic environments, Real-Time A* (RTA*) [79] [80] compromises optimality with feasibility, and defines a maximum time threshold for each iteration from which the algorithm mandatorily advances to the next step. Lifelong Planning A* (LPA*) [81] and D*[82] take into account cost changes in the graph re-using previous A* computations to find the new optimal path without starting from scratch. Those algorithm usually integrate a sort of α-pruning technique that considers a variable called α containing the least computed cost such that any node whose cost is bigger than α threshold is never expanded. Jump Point Search (JPS) [83] in cost-uniform graphs is a technique that relies on search space pruning to reduce considerably computation times. Iterative-Deepening A* (IDA*)[84] algorithm combines the best of iterative deepening and A* to reduce space complexity (memory requirements), and initially performs the search in depth cutting a branch if a threshold is surpassed and maintains A* completeness and optimality. Occasionally, bidirectional search algorithms [85] has demonstrated to reduce computation effort but, in the worst cases, they can even perform worse than unidirectional search.

Finally, reinforcement learning and probabilistic analysis have converged for solving problems with huge state spaces - e.g. board games such as chess or go - in which a value function, usually improved over time, prioritizes the best candidate actions with regards to an optimal policy. Related work has been done with algorithms such as Monte Carlo Tree Search (MCTS) [86] [87] or X-armed bandits agents [88] that rely on plenty of look-ahead simulations with the aim of backpropagating the results, improve the policy and make better decisions in the future. In the scope of this study, that would mean to compute plenty of trajectories for many arrival procedures, the policy is improved over time and the algorithm would choose the actions according to their experience in similar situations. Despite being a very interesting research axis, stochastic processes govern these approaches; hence generalization and certification issues may become the function

impractical for critical on-board purposes.

2.8 Conclusion

This chapter has presented the general functioning principles of the FMS and some design points that could improved. The use of the FMS in descent and approach phases puts in evidence that the workload of flight crews during these phases of flight is high. This is especially evident when aircraft deviate from their intended paths, since the trajectory-related information provided by the FMS does not correspond to that of the current aircraft state. Hence, flight crews define a flight strategy to manage the aircraft energy efficiently and ensure the safety of the flight. In a second time, the trajectory optimization problem to be solved, which is frequently considered as an OCP, has been presented. The goal of these type of problems is to find the sequence of flight control actions that generate a certain number of states which are part of the optimal trajectory. In the literature, most researches have solved OCPs through direct or indirect methods, although other methodologies have also been applied. In this thesis, early trials using indirect methods have suggested that constraints are difficult to handle, since the derivation of the Hamiltonian is difficult to obtain. Moreover, direct methods do not yield the global optima solution, which highly depends on the initial guess. These reasons have lead to the conclusion that DP could provide efficient techniques to solve clustered environments, as it is the case of trajectory optimization in descent and approach phases. In particular, the A* algorithm finds the global optima when the heuristic is well defined. Variants of A* algorithm presented at the end of the chapter enlarge the perimeter and could be very useful for certain purposes, for instance, when the algorithm needs to provide a solution in a given time or when part of the previous solution is reused for tactical adjustments. In the next chapter, the complete mathematical formulation of the trajectory optimization problem is given. Besides, two performance models are proposed to compute the optimal trajectories and the hypotheses used by the algorithm to compute the optimal trajectory are presented as well.

3

Mathematical Model

This chapter presents the mathematical formulation of the problem. First, aircraft equations of motion are presented. Then, the constraints of the problem are formulated and both Airbus FMS and BADA performance models are presented. The chapter finalizes with the presentation of the optimal control formalism and the objective function to optimize.

3.1 Aircraft Equations of Motion

Aircraft vertical motion is usually represented through one of these three different models: rigid-body, point-mass or energy-state [89]. The rigid-body model is the more representative but requires large computation times with respect to energy-state models, since more variables are involved in the calculations. Instead, point-mass models provide a sufficient level of representativeness while being simple and reducing the computational effort. This is probably the reason why most works in the literature use this type of model for computing aircraft trajectories. Moreover, rigid-body models are generally used when fast-dynamics of the aircraft are relevant, as it is the case for calculations related to handling qualities and flight controls laws design. Trajectory generation focuses on slow dynamics variables, and a point-mass model represents a sufficient level of detail for this purpose. Since this thesis focuses on the generation of trajectories and the definition of guidance laws to follow such trajectories is out of the scope of the thesis, the point-mass model provides an appropriate representation. The general hypotheses concerning to the mathematical formulation adopted in this thesis are listed hereinafter:

- Gravitational acceleration (g_0) changes due to geometric height are considered negligible.
- Ground distance s is considered as the independent variable instead of time t.
- Thrust force is collinear with the speed vector.
- The earth curvature is neglected as its impact in short-distance phases such as descent and approach is low.
- The fast dynamic of Flight Path Angle (FPA) ($\dot{\gamma} = 0$) is disregarded.
- Wind influence is limited to the horizontal component (V_w).

3.1.1 Time-dependent equations of motion

The time-dependent equations of motion for a point-mass aircraft in the vertical plane are defined by the following expressions:

$$\begin{cases} \dot{s} = \dfrac{ds}{dt} = V\cos\gamma + V_w \\ \dot{h} = \dfrac{dh}{dt} = V\sin\gamma \\ \dot{V} = \dfrac{dV}{dt} = \dfrac{Thr - D - mg_0\sin\gamma}{m} \\ \dot{m} = \dfrac{dm}{dt} = -FF \end{cases} \tag{3.1}$$

The detail of these equations can be found in appendix A. Here, the independent variable is time, whose final value is unknown. Most constraints are distance-defined and the final distance is known as it corresponds to the aircraft position. Thus, the time-dependent equations of motion can be converted into distance-dependent by means of the following term:

$$\frac{dt}{ds} = \frac{1}{V\cos\gamma + V_w} \tag{3.2}$$

3.1.2 Distance-dependent equations of motion

The wind component function of the altitude in Eq.(3.1) is neglected because its influence is limited in relatively short segments. The term defined in (3.2) converts the time-dependent equations of motion into distance-dependent, which results in the following formulation:

$$\begin{cases} h' = \dfrac{dh}{ds} = \dfrac{V\sin\gamma}{V\cos\gamma + V_w} \\ V' = \dfrac{dV}{ds} = \dfrac{Thr - D - mg_0\sin\gamma}{m\,(V\cos\gamma + V_w)} \\ m' = \dfrac{dm}{ds} = \dfrac{-FF}{V\cos\gamma + V_w} \\ t' = \dfrac{dt}{ds} = \dfrac{1}{V\cos\gamma + V_w} \end{cases} \tag{3.3}$$

It is noted that the prime symbol denotes the derivation with respect to the distance, which is the independent variable. It is defined as the traveled distance flown by the aircraft or ground distance. The justification behind this choice is because the initial and final distance are known, as they correspond to the stabilization point and the aircraft position respectively, whereas the final time is unknown.

The aerodynamic drag force (D) is defined by:

$$D = \frac{1}{2}\rho V^2 S C_D \tag{3.4}$$

Where ρ is the air density, V the true airspeed, S the wing surface and C_D the drag coefficient. The lift coefficient (C_L) in the vertical plane is computed taking into account the FPA (γ) and the bank angle (ϕ):

$$C_L = \frac{2mg_0 \cos\gamma}{\rho V^2 S \cos\phi} \tag{3.5}$$

Aircraft energy state is defined as the sum of potential and kinetic energy:

$$E_T = E_p + E_k = mg_0 h + \frac{1}{2}mV^2 \tag{3.6}$$

Specific total energy or energy height (E_{T_s}) can be obtained dividing Eq. (3.6) by the aircraft weight (mg_0):

$$E_{T_s} = E_{p_s} + E_{k_s} = h + \frac{1}{2}\frac{V^2}{g_0} \tag{3.7}$$

The derivation of Eq. (3.7) yields the energy rate:

$$\dot{E}_{T_s} = \dot{E}_{p_s} + \dot{E}_{k_s} = \dot{h} + \frac{V\dot{V}}{g_0} \tag{3.8}$$

The ESF is defined as the percentage of the total energy rate dedicated to the kinetic energy:

$$ESF = \frac{\dot{E}_{k_s}}{\dot{E}_{k_s} + \dot{E}_{p_s}} \tag{3.9}$$

The combination of (3.8) and (3.9) with the differential equation $\dot{h} = V \sin\gamma$ gives the following expression:

$$\sin\gamma = \frac{(1 - ESF)}{ESF}\,\frac{\dot{V}}{g_0} \tag{3.10}$$

In order to simplify the equations and adopt an aircraft energy-state representation, the total flight path angle (γ_T) is defined as the sum of the aerodynamic flight path angle and the corresponding acceleration, as shown in Fig. 3.1.

$$\sin\gamma_T = \sin\gamma + \frac{\dot{V}}{g_0} = \frac{\sin\gamma}{1 - ESF} = \frac{(Thr - D)}{mg_0} \tag{3.11}$$

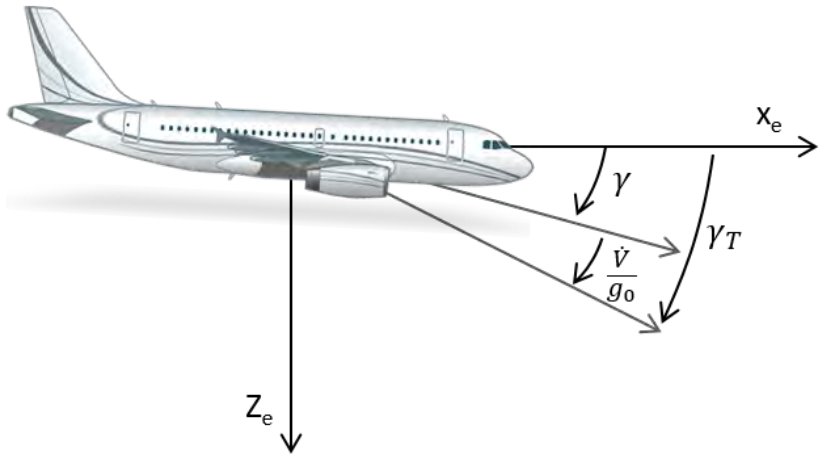


Figure 3.1: Representation of total flight path angle (γ_T).

Considering that γ is relatively small so that $\sin\gamma \approx \gamma$ and $\sin\gamma_T \approx \gamma_T$, the combination of Eqs. (3.11) and (3.9) with aircraft equations of motion Eq. (3.3) results in the following formulation:

$$\begin{cases} h' = \dfrac{V\ (1 - ESF)\ \gamma_T}{V\cos\gamma + V_w} \\ V' = \dfrac{g_0\ ESF\ \gamma_T}{V\cos\gamma + V_w} \\ m' = \dfrac{dm}{ds} = \dfrac{-FF}{V\cos\gamma + V_w} \\ t' = \dfrac{dt}{ds} = \dfrac{1}{V\cos\gamma + V_w} \end{cases} \tag{3.12}$$

The equations of motion described in Eq. (3.12) results in a state vector x of the form:

$$\text{state variables} \rightarrow x(s) = [h, V, m, t] \tag{3.13}$$

Whereas the control variables u of the associated optimal control formulation are defined by the following vector :

$$\text{control variables} \rightarrow u(s) = [\gamma_T, ESF, \delta_{ab}, \text{Conf}] \tag{3.14}$$

The variables δ_{ab} and Conf do not appear explicitly in Eq.(3.12). They are implicitly considered as airbrakes deflection (δ_{ab}) increases the drag force whereas flap configuration changes (Conf) increase the drag force and also slightly the idle thrust. Alternatively, γ_T and ESF could have been replaced by Thr and γ as control variables and the result would

be the same. However, γ_T and ESF provide a concise and more representative formulation of the aircraft energy state, since γ_T refers to the excess or loss of energy and ESF defines the way energy is distributed. This choice enables to directly relate altitude and speed as both control variables appear in both altitude and speed differential equations (3.12) as opposed to (3.3) where Thr only appears in the speed equation.

3.2 Performance Models

3.2.1 Navigation Database (NDB)

Constraints on state variables are dictated by arrival procedures design. They are contained in the navigation database (NDB) under the form of altitude and speed constraints, coded in ARINC 424 [90] standard by database providers. They impose that aircraft altitude shall remain below, above, in-between or at a certain flight level when it applies:

$$\begin{cases} \text{AT OR ABOVE} \rightarrow \quad h \geq h_{CSTR} \\ \text{AT OR BELOW} \rightarrow \quad h \leq h_{CSTR} \\ \text{WINDOW} \rightarrow \quad h_{CSTR} \geq h \leq h_{CSTR} \\ \text{AT} \rightarrow \quad h = h_{CSTR} \end{cases} \tag{3.15}$$

A speed constraint restricts aircraft Calibrated airspeed (CAS) below a certain speed:

$$V_{CAS} \leq V_{CAS_{CSTR}} \tag{3.16}$$

Furthermore, ATC regulation generally imposes a maximum speed of 250 kt CAS for all aircraft below FL100:

$$V_{CAS} \leq V_{CAS_{SPDLIM}} \quad \forall h \leq \text{FL100} \tag{3.17}$$

Finally, aircraft shall not fly below the glide slope beam on final approach in order to ensure obstacle clearance and comply with current operations (see Fig. 3.2).

$$\gamma \leq -3^{\text{o}}, \quad \forall s \in \text{glide path} \tag{3.18}$$

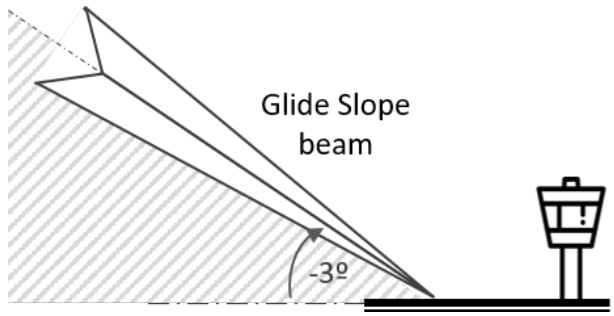


Figure 3.2: Glide Slope beam limitation on final approach.

3.2.2 Performance Database (PDB)

Control variables γ_T and ESF are bounded between a maximum and minimum value; thrust power limits the total flight path angle whereas ESF is operationally limited for passenger comfort purposes:

$$\gamma_{T_{min}}\Big|_{Thr=Thr_{idle}} \leq \gamma_T \leq \gamma_{T_{max}}\Big|_{Thr=Thr_{max}} \tag{3.19}$$

Climb segments ($\gamma > 0$) are not allowed during descent phase so any $\gamma_T > 0$ corresponds to a gain of kinetic energy. The ESF distributes the energy budget between the kinetic and the potential energy. For idle segments, the value is bounded between:

$$ESF \in \{-0.5, ..., 1\} \tag{3.20}$$

For non-idle segments, the ESF only takes the value 1, i.e. the energy gain ($\gamma_T > 0$) is dedicated to accelerate the aircraft. To illustrate the impact of the energy share, considering idle thrust, an $ESF = 1$ leads to a decelerated level-flight, since γ can be defined as a function of γ_T and ESF:

$$\gamma = \gamma_T \cdot (1 - ESF) \tag{3.21}$$

In that case, γ equals to zero and the aircraft is in level flight. $ESF < 0$ represents an increase of kinetic energy as a consequence of a potential loss (steep descent). A value of $ESF = 0$ keeps aircraft true-airspeed constant as it descends. Hence, constant V_{CAS} segments require a part of the total energy loss to decrease V as the aircraft descends.

The extension of airbrakes has an impact on aerodynamic drag proportional to the deflection angle (δ_{ab}). Airbrakes extension is limited to three lever positions to be consistent

with current lever design:

$$\delta_{ab} \in \{0, \delta_{ab_{half}}, \delta_{ab_{full}}\} \tag{3.22}$$

In addition, aircraft airspeed shall remain within the flight envelope for clean configuration that is defined by the stall speed (V_{LS}) and the maximum operating speed (V_{MO}):

$$V_{LS} \leq V_{CAS} \leq \ V_{MO} \tag{3.23}$$

In high-lift configuration, the upper bound is defined by the maximum flaps extended speed (V_{FE}) and the minimum speed V_{LS}, which decreases as soon as flap configurations are set:

$$V_{LS} \leq V_{CAS} \leq \ V_{FE} \tag{3.24}$$

Passengers comfort is also taken into account by limiting the longitudinal acceleration to a certain g-force value [91]:

$$|\dot{V}| \leq 0.06 \ g_0 \tag{3.25}$$

3.2.3 Operational Constraints

The previously set of constraints comply with the arrival procedure design and aircraft flight envelope but do not take into account the operational feasibility of the resulting trajectory. Hence, additional constraints on state and control variables would result in a meaningful trajectory from an operational perspective. A kinetic energy increase as a result of a potential energy decrease, is forbidden below 8000 ft, since combined with a strong tailwind, could provoke an undesirable over-speed at altitudes where the surrounding traffic is dense.

$$ESF \in \{0.1, ..., 1\} \quad \forall h \leq \text{FL80} \tag{3.26}$$

In order to avoid long descent times, aircraft minimum speed is limited to 250 knots above FL100.

$$V_{CAS} \geq V_{CAS_{SPDLIM}} \quad \forall h \geq \text{FL100} \tag{3.27}$$

The operational constraint defined in (3.27) is enabled only for certain procedures, in compliance with published speed constraints. It shall be disabled in those procedures requiring low speeds above FL100, otherwise the algorithm would not find any solution. Finally, the extension of airbrakes increases V_{LS} so their usage is inhibited for flaps configuration 3 and full:

$$\delta_{ab} = 0, \quad \forall Conf \in \{3, \text{full}\} \tag{3.28}$$

3.3 Airbus FMS Performance Model

A performance computational model has been developed and is used by the algorithm to generate the optimal trajectories. Airbus PDB contains engine, aerodynamic and aircraft performance data stored under the form of labelled lookup tables (see Fig.3.3). Each aircraft version for each engine type has its own PDB, which is directly loaded into the airplane and used by the FMS to perform the calculations. Hence, the model developed for this thesis is based on the actual FMS model and uses the same PDB as the actual system. However, since performance data is industrial property of Airbus, their values cannot be disclosed and only the relations will be depicted.

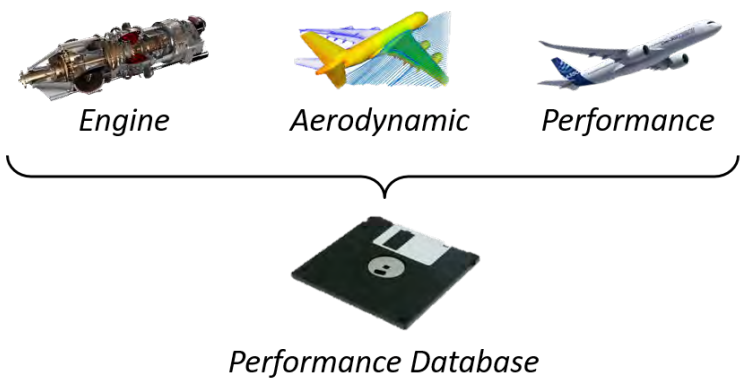


Figure 3.3: Models contained in a PDB

Regarding to the engine model, a scaling factor called Thrust Setting Parameter (TSP) is bounded between a minimum and maximum value:

$$TSP_{max/min} = f_1(h, M) \tag{3.29}$$

Minimum value is used for idle segments whereas maximum TSP is applied in climb phase. These TSP bounds are used for computing the minimum and the maximum thrust for a given altitude and speed:

$$Thr_{max/min} = f_2(h, M, TSP) \tag{3.30}$$

For non-idle segments (also known as geometric), thrust is calculated from equations of motion (3.3) and then, TSP is iterated from equation (3.29). Hence, fuel consumption is computed through the following expression:

$$FF = f_3(h, M, TSP) \tag{3.31}$$

Finally, drag is calculated from the following state and control variables:

$$D = f_4(h, M, Conf, \delta_{ab}) \tag{3.32}$$

3.4 Performance Computational Model Validation

In order to quantify fuel and time savings, the computational model inserted in the advanced A* algorithm has to be representative of that used by the FMS. Otherwise, it would be difficult to analyze and quantify precisely any fuel or time savings. A tool developed by Airbus flight performance teams, called PSIMU, has been used for the validation of the developed model, since it is representative of a real FMS software. This tool enables to generate a full vertical profile based on a set of user inputs, which can then be compared with the same trajectory integrated in Matlab based on A* performance computational model. The resulting path is composed of a set of segments integrated backwards using FMS hypotheses. For this case study, ten segments constitute the vertical profile between the runway threshold and the cruise flight level as shown in Fig. 3.4, which are described hereinafter:

1. Geometric segment at approach speed (V_{APP}) and full aerodynamic configuration, which starts at runway threshold and finish at stabilization point (1000 ft above runway, full configuration).
2. Acceleration at -3 FPA from stabilization point up to next characteristic speed, where a change of configuration occurs.
3. Since the aircraft decelerates smoothly in configuration 2 with a FPA of -3º, glide-slope FCA at 4000 ft is achieved before a change of conf occurs.
4. In level-flight, aircraft accelerates to the next characteristic change speed (conf 2 to 1).
5. In level-flight, aircraft accelerates to the next characteristic change speed (conf 1 to clean).
6. In level-flight, aircraft accelerates to next speed target, which is 250 knots as imposed by ATC speed limitation (see SPD LIM). It delimits the transition between approach and descent path where deceleration pseudo-waypoint is localized.
7. Descent at constant target speed up to FL100 (SPD LIM ALT).
8. Acceleration to descent economic speed, OPT SPD, (typically 250 knots for CI=0) with a fixed ESF.

9. Descent at ECON DES speed up to crossover altitude (XOVER ALT), which is defined as the altitude at which the economic Mach and CAS couple has the same true airspeed.

10. Descent at constant Mach, producing a deceleration in CAS up to the cruise flight level.

The concatenation of those segments define the theoretical descent and approach profile that will be followed by guidance if autopilot is engaged.

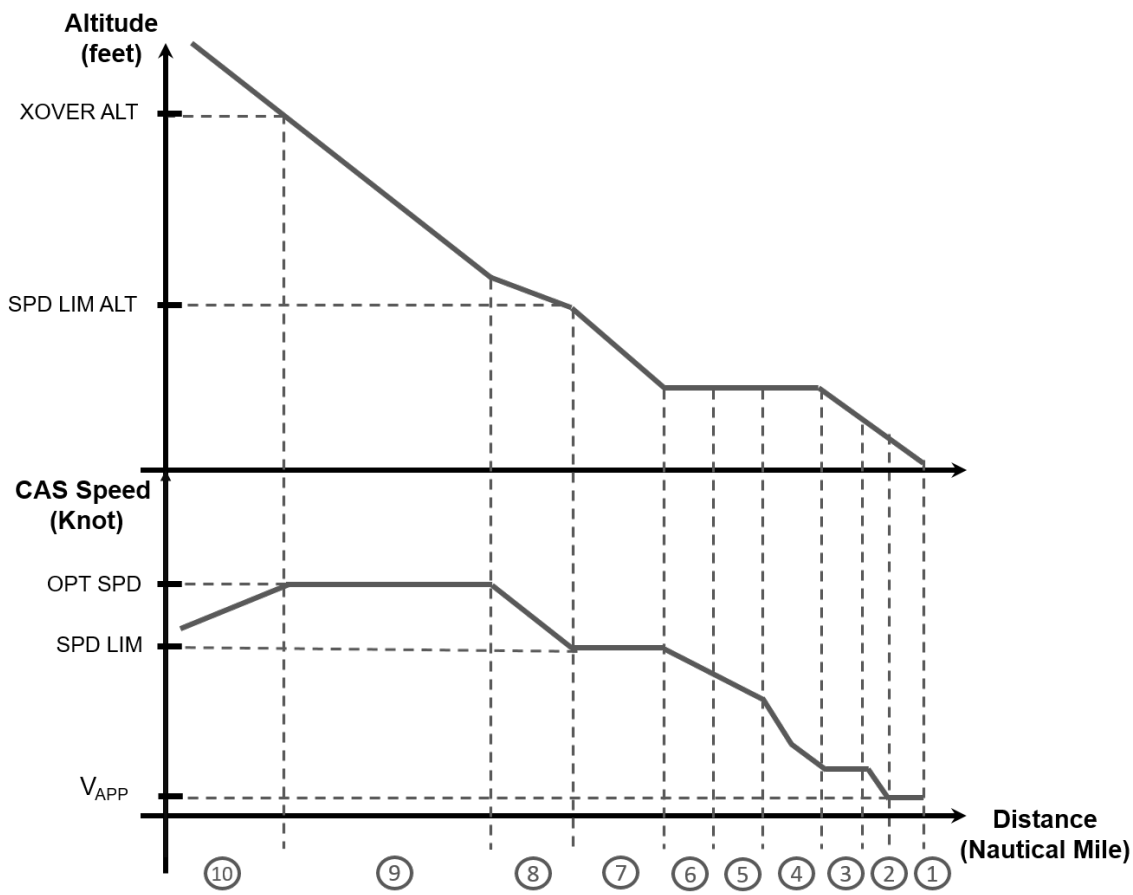


Figure 3.4: Altitude (upper) and speed (bottom) profile.

The previous altitude and speed profiles depicted in Fig. 3.4 have been computed by the A* model in order to be compared to those generated by the Airbus PSIMU. The goal is to validate the model with PSIMU software to ensure that the A* model does not contain errors. The PSIMU tool has been developed by the flight performance team and is based on the real FMS model, whose purpose is to compute vertical profiles for comparison.

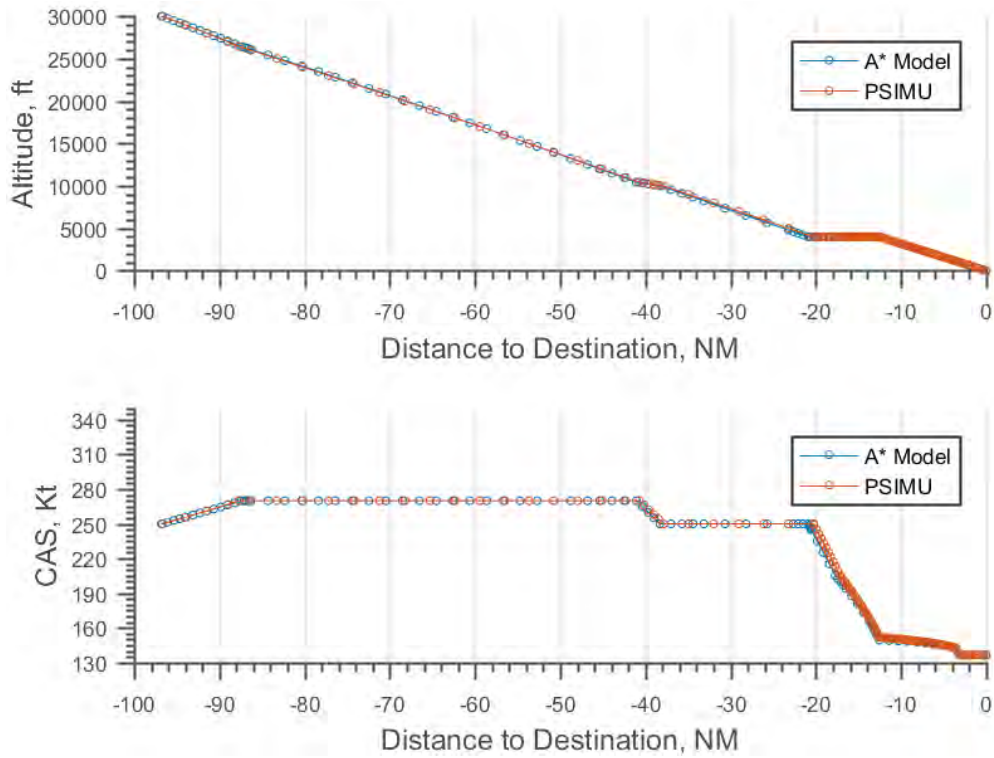


Figure 3.5: Profile comparison between PSIMU and A* developed model.

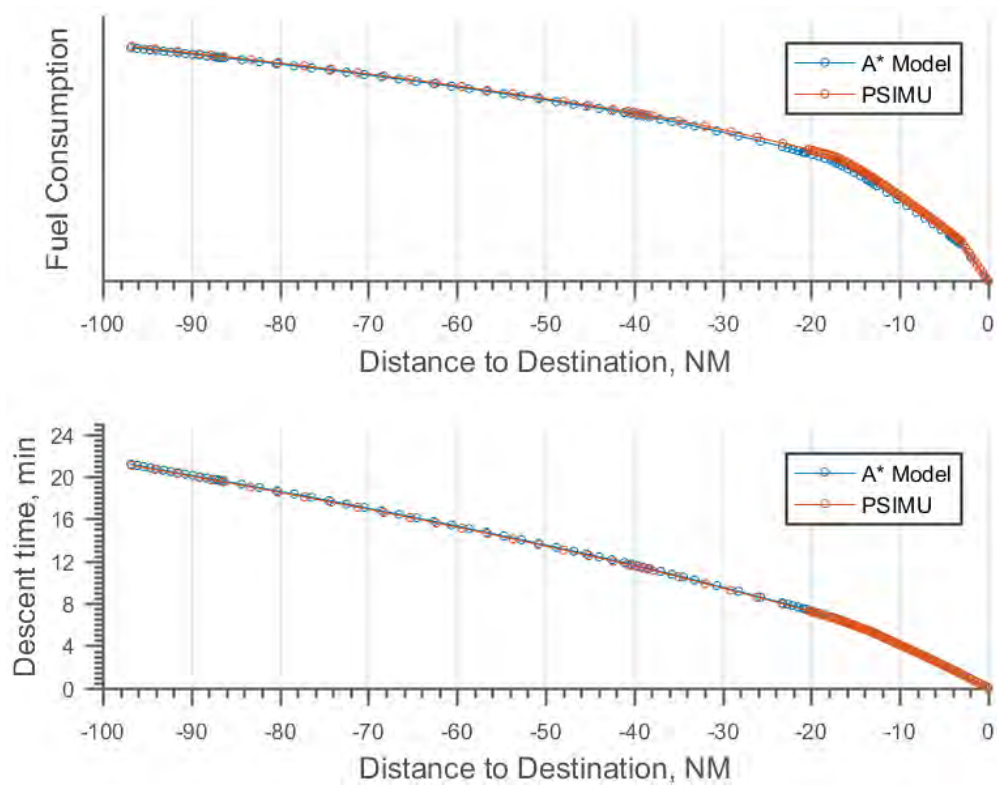


Figure 3.6: Fuel and time comparison between PSIMU and A* developed model.

The vertical profile is displayed in Fig. 3.5 while the flight performance associated to the profile is displayed in Fig. 3.6. It can be observed from Fig. 3.6 that fuel consumption difference is less than 0.5%. It has to be noted that, due to confidentiality issues, fuel consumption values are not displayed. A minor difference in the deceleration capability during approach is observed but does not impact the quality of the solution. This deviation may be related to the integration method applied. The previous results suggest that the developed model used in the algorithm is representative of that of a real FMS.

3.5 BADA Performance Model

Optimal trajectories can also be computed using BADA model [91], since Airbus model contains proprietary sensitive information, allowing broader distribution of results. This model is based on a total-energy model that is defined through:

$$(Thr - D)V = mg_0 \frac{dh}{dt} + mV \left(\frac{dV}{dh}\right)\left(\frac{dh}{dt}\right) \tag{3.33}$$

In this model, engine thrust for each flight phase is a percentage of the maximum climb power, which is dependent of aircraft altitude:

$$(Thr_{maxclimb})_{ISA} = C_{Tc,1} \cdot \left(1 - \frac{h}{C_{Tc,2}} + C_{Tc,3}\ h^2\right) \tag{3.34}$$

Where $Thr_{maxclimb}$ is the maximum climb thrust at standard atmosphere conditions whereas $C_{Tc,1}$, $C_{Tc,2}$ and $C_{Tc,3}$ are values specific to the engine type. Taking into account atmospheric deviations with respect to the international atmosphere model:

$$Thr_{maxclimb} = (Thr_{maxclimb})_{ISA} \cdot (1 - C_{Tc,5} \cdot \Delta T_{eff}) \tag{3.35}$$

The corrected maximum climb thrust is given by $Thr_{maxclimb}$, $C_{Tc,5}$ is a constant value and ΔT_{eff} is a parameter that depends on the ambient conditions:

$$\Delta T_{eff} = \Delta\text{ISA} - C_{T_{c,4}} \tag{3.36}$$

Where $C_{T_{c,4}}$ is another constant value of the engine. The corrected maximum climb thrust is given by $Thr_{maxclimb}$ and $C_{Tc,5}$ is a constant value. Descent thrust is calculated

from the maximum climb thrust given by Eq.(3.35):

$$Thr_{des/high} = C_{Tdes/high} \cdot Thr_{maxclimb} \tag{3.37}$$

$$Thr_{des/high} = C_{Tdes/low} \cdot Thr_{maxclimb} \tag{3.38}$$

$$Thr_{des/app} = C_{Tdes/app} \cdot Thr_{maxclimb} \tag{3.39}$$

$$Thr_{des/ld} = C_{Tdes/ld} \cdot Thr_{maxclimb} \tag{3.40}$$

The constants $C_{Tdes/high}$, $C_{Tdes/low}$, $C_{Tdes/app}$ and $C_{Tdes/ld}$ are defined according to the phase of flight: descent above a reference altitude, descent below that altitude, approach and landing. Then, fuel consumption is defined as a function of aircraft thrust and true-airspeed:

$$\eta = C_{f1t}\left(1 + \frac{V}{C_{f2}}\right) \tag{3.41}$$

In Eq.(3.41), C_{f1t} and C_{f2} are thrust specific fuel consumption coefficients.

$$f_{nom} = \eta \cdot Thr \tag{3.42}$$

For idle segments, the minimum fuel consumption is calculated as a function of altitude:

$$f_{min} = C_{f3}\left(1 - \frac{h}{C_{f4}}\right) \tag{3.43}$$

Where C_{f3} and C_{f4} are coefficients of the fuel model. Hence, fuel flow for descent and approach phases is the maximum between both values:

$$f_{ap/ld} = max(f_{nom}, f_{min}) \tag{3.44}$$

In clean configuration, the drag coefficient (C_D) is computed from the drag polar curve, which is defined as the sum of parasitic and inductive drag:

$$C_D = C_{D0,CR} + C_{D2,CR} \cdot {C_L}^2 \tag{3.45}$$

Where $C_{D0,CR}$ is the parasitic drag coefficient and $C_{D2,CR}$ is the induced drag coefficient. In landing configuration, flaps are sequentially extended until decelerating the aircraft to approach speed with an increase of drag coefficient:

$$C_D = C_{D0,APPR} + C_{D0,\Delta LDG} + C_{D2,APPR} \cdot {C_L}^2 \tag{3.46}$$

The parasitic ($C_{D0,APPR}$) and induced drag ($C_{D2,APPR}$) are specific to the approach phase and depend on the flap configuration whilst $C_{D0,\Delta LDG}$ is added as soon as the gear is down. The effect of airbrakes on drag (D) is taken into account through the expedite descent multiplication factor ($C_{D,exp}$) that increases nominal drag:

$$D = C_{D,exp} \cdot D_{nom} \tag{3.47}$$

3.6 Optimal Control Theory

OCP applied to trajectory optimization are generally defined through a set of ordinary differential equations of the form:

$$x'(s) = f\Big(x(s), u(s), s\Big) \tag{3.48}$$

In this case, the independent variable is distance instead of time as the final distance s_f is known whereas t_f is unknown. The state vector is defined by $x(s)$, $u(s)$ defines the control and s the distance. The problem is subjected to a set of non-linear boundary conditions of the form:

$$g\Big[x(s), u(s), s\Big] \leq 0 \tag{3.49}$$

while the performance index to be optimized is defined as:

$$J = \min \int_{s_0}^{s_f} \mathcal{L}\Big[x(s), u(s)\Big] \, ds \tag{3.50}$$

This thesis focuses on the minimization of the flight cost for descent and approach phases, which is defined as the sum of time and fuel consumption along the trajectory:

$$J = \min \int_{s_0}^{s_f} \left(FF + \frac{\mathrm{CI}}{60}\right) \frac{1}{V\cos\gamma + V_w} \, ds \tag{3.51}$$

FF being expressed in $\frac{Kg}{s}$. The Cost Index is generally defined as the ratio between time (C_t) and fuel (C_f) cost:

$$\mathrm{CI} = \frac{Ct}{C_f} \tag{3.52}$$

which is expressed in $\frac{\$/min}{\$/Kg} = \frac{Kg}{min}$. In general, airlines choose a CI ranging between 10-30, which prioritizes the optimization of fuel instead of time. Hence, setting CI = 0

restricts the cost function to the optimization of the fuel burnt:

$$J = \min \int_{s_0}^{s_f} \left(FF\right) \frac{1}{V\cos\gamma + V_w} \, ds \tag{3.53}$$

On the contrary, time optimization can be of interest to minimize delays and passenger compensations resulting from missing their connecting flight, so for sufficiently large CI values, the fuel term can be negligent:

$$J = \min \int_{s_0}^{s_f} \left(\frac{\mathrm{CI}}{60}\right) \frac{1}{V\cos\gamma + V_w} \, ds \tag{3.54}$$

3.7 Conclusion

The complete formulation of the trajectory optimization problem for descent and approach phases of a commercial aircraft have been presented in this chapter. This includes two performance models with which the algorithm can generate trajectories; the first one is based on a real Airbus FMS model while the second is based on BADA model. The former model was compared with the Airbus PSIMU tool, which validates that the model does not contain errors. The BADA model has been coded as it is, although it has not been compared with any other software to properly assess its validity. The set of constraints are presented in a general way and may not be applicable depending on the selected STAR and Approach Route (APPR) procedure, since each procedure has its unique design. The optimal control formalism introduced in the last part of the chapter presents the objective function to be minimized. The implemented methodology for solving the OCP is presented in detail in the next chapter, which is a particular version of a dynamic programming algorithm.

4

Optimal Arrival Trajectories with A* Algorithm

This chapter details the functioning of the algorithm that is applied to the computation of optimal trajectories in the vertical plane for commercial airplanes. The proposed algorithm is a customized version of A [92], which applies dynamic programming to solve the optimal control problem. The chapter describes the design of the algorithm with a special focus on the gradually generation of the search space. The chapter is completed with the study of the heuristic function, whose implementation highly influences the performance and quality of the solution.*

4.1 The A* algorithm

The well-known A* algorithm [74] is a path-finding algorithm that finds the shortest path on a graph. The algorithm prioritizes nodes according to a score (f-cost). This score is the result of the sum of the path-to-go to the current node, g-cost, and an estimate of the remaining cost from the current node to the target node (h-cost). The node with the least f-cost is developed and the computation finishes as soon as the least f-cost is the target node. The A* algorithm is complete, as it finds the optimal solution if it exists, and ensures that the solution is global optimum as long as the heuristic function satisfies the admissibility property. A heuristic function is admissible as long as it always under-estimates the optimal cost for each node of the graph. The heuristic function shall be defined properly to ensure the optimality of the solution and the efficiency of the algorithm. The closer the heuristic is to the optimal cost, the fewer nodes the algorithm will explore [93]. In reality, A* is optimally efficient as there is no other optimal algorithm that guarantees to develop fewer nodes than A* when the heuristic is admissible [94].

The A* algorithm has already been implemented to trajectory optimization problems in works such as [95], [96], [50],[97] and [49], either with the aim to compute end-to-end trajectories or particular flight phases. Most of these applications generate a priori the search space in which the A* algorithm finds the optimal path. The algorithm described in this thesis differs from the traditional A* implementation in several aspects. On one hand, the graph is gradually constructed instead of being pre-processed, because the flight conditions may change at any time and that requires the re-computation of the whole search space. For continuous re-computations, part of the previous nodes can be reused by computing again their f-cost. On the other hand, the construction of the search space is tailored to the particular problem to solve. The constraints of the problem prune the search space and help to generate nodes at the right locations. The objective is to minimize the impact of the control variables discretization and to compute the optimal trajectory independently of the constraints location. This tailored construction is one of the main contributions of this work, since it generalizes the algorithm to compute the optimal trajectory for any given arrival procedure. Moreover, the algorithm adapts the control variables of the problem according to the flight phase and the aircraft energy condition.

4.2 Implementation of A* to trajectory optimization

The present thesis proposes a customized version of the A* algorithm to the computation of optimal trajectories for descent and approach flight phases. As a reminder, the main advantage of this algorithm is that it finds the global optimal of the problem whenever the

heuristic function satisfies a few properties. However, the performance of the algorithm is dependent on the quality of the heuristic and the size of the problem, i.e. the combinatorics of the problem. The following schema provides a general view on the functioning of the algorithm:

Algorithm 1 Schema of the A* algorithm implementation.

```
Initial search space generation and pruning
Initiate Open (𝒪) and Closed (𝒞) list
while node ≠ Target do
    Generate children nodes from current

    if node ∈ 𝒞 then
        Discard node;
    else
        Insert node in 𝒪
        if node ∈ 𝒪 then
            Compare g-costs
            Increase node priority if necessary
        else
            Insert node in 𝒪
        end
    end
    Best-node removal from 𝒪
    Insert node in 𝒞
    Set node as current
end
Retrieve parents from target until initial node
Plot results
```

The search space initially only contains the initial and target nodes and a set of constraints that are located at a certain distance of the runway threshold. The generation of nodes starts from the initial node and terminates at the target node. For each node generated, the algorithm checks if it is already in the closed list in which case the node is removed as it has been already visited. On the contrary, the node may be in the open list, then g-costs are compared and only the node with the lowest value is maintained. New nodes are added to the open list and the one with the lowest f-cost is retrieved from the list to continue with the process. That node becomes the current one, from which

new nodes are generated. The algorithm terminates when the lower f-cost node is the target itself. Then, the optimal trajectory is obtained through pointers to parent nodes until reaching the initial node. The general functioning is analogue to the standard A* algorithm, however, the design of each function has been customized to the problem to solve as it is detailed in next sections.

4.3 Search space generation

The representation of the search space usually depends on the problem to solve; certain problems pre-process a map where the algorithm seeks the optimal path whereas others require to generate progressively such map. This is the case of trajectory optimization problems where additional constraints are generally added on top of those already existing. This is why pre-processed maps would require to store the search space in a database updated according to the Aeronautical Information Regulation And Control (AIRAC) cycle. The NDBs are updated and uploaded on the aircraft every 28 days in order to ensure that all operators use the same type of navigation information. For this purpose, it seems appropriate to generate the nodes for each prime calculation and then re-use some parts of the graph in the case that any readjustment is required. As a consequence, the size of the search space is variable and depends on the number of discrete variables, constraints and the quality of heuristics.

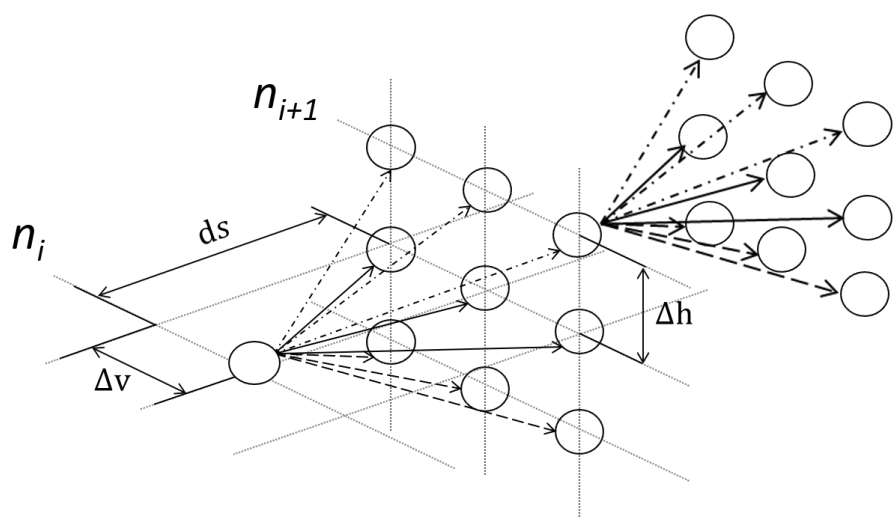


Figure 4.1: A* progressive search space generation.

The calculation of the optimal trajectory is performed upstream from the stabilization point up to the cruise level. The algorithm launches the calculation from the initial node as displayed in Fig. 4.1; children nodes are generated and only the one with the lowest

f-cost is developed and put in the closed list. The process goes on until the less costly node is the target node itself, meaning that the optimal path has been found. At that point, pointers to each child node are used for retrieving the optimal trajectory.

4.4 Search space pruning

The search space of the problem is defined by a graph of the form $\mathcal{G}(e, v)$, where e are the edges of the graph that define a trajectory linking any pair of vertexes, v, commonly known as nodes or aircraft states. At the end of the calculation, as shown in Fig. 4.2, the search space contains a set of generated nodes (black) of which a few have been both developed (gray-crossed) and pertain to the optimal path (red-crossed). The projection of optimal nodes on the XY-plane shapes the speed profile (green line) whereas the projection on the XZ-plane yields the altitude profile (blue line). Altitude and speed constraints are represented through pink-colored triangles.

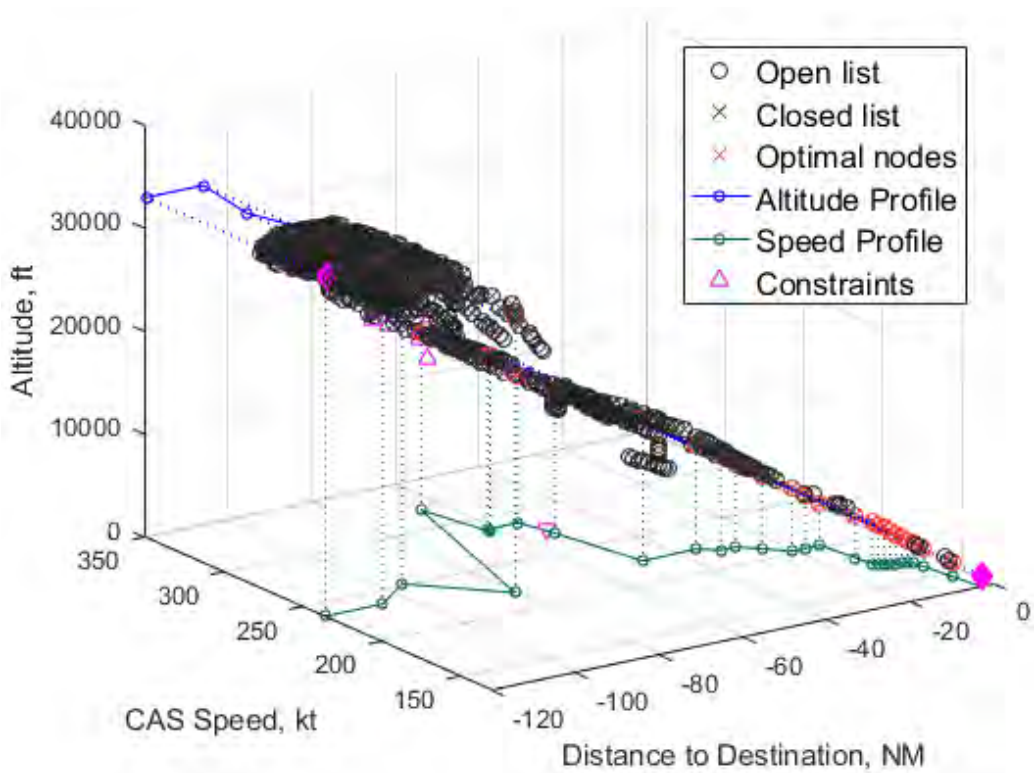


Figure 4.2: The search space at completion of the algorithm run.

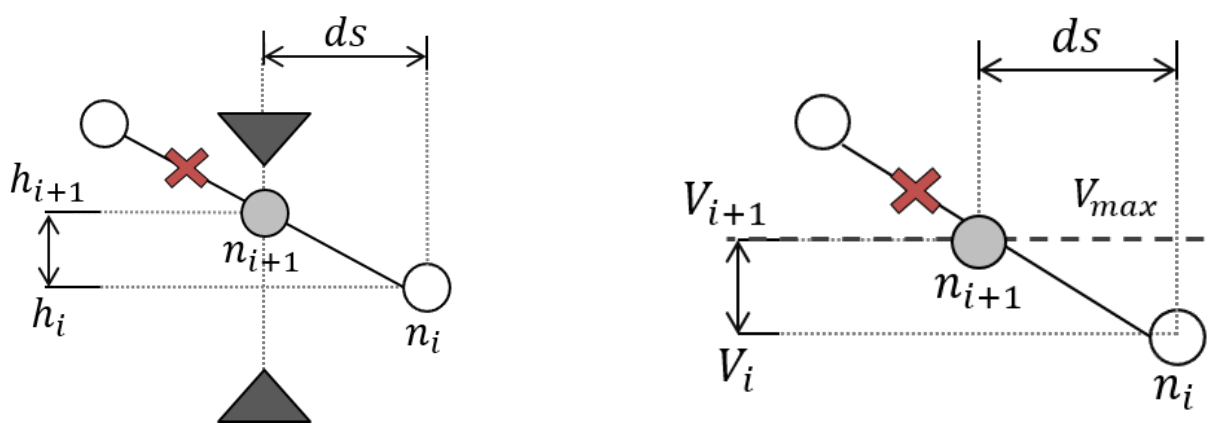
The size of the search space may be considerable and any knowledge of the problem is used for reducing the number of combinations. For each iteration, the position of the current node can discard some control variables, for instance, a current node with maximum speed can only decrease its speed in the same way that a node at the AT or

BELOW altitude can only perform a level-off. Similarly, acceleration and decelerations limits are used for pruning the area around the target node. Other parts of the search space are basically non-interesting and shall be avoided during the calculation. Thus, knowledge of the problem reduces the number of combinations and ultimately the candidate nodes, so the performance of the algorithm is improved.

Each node of the graph defines an aircraft state that is a candidate for being part of the optimal trajectory. Nodes are defined by the state vector $x(s) = \{h, V, m, t\}$ whilst aircraft states are generated by means of the controls variables defined by $u(s) = \{\text{ESF}, \gamma_T, \delta_{ab}, \text{Conf}\}$. Therefore, the time-complexity of the algorithm is dependent on the number of combinations and candidate nodes generated.

4.5 Node generation process

The calculation starts with an initial and target node and the discretization of the control variables with a distance step (ds) which is initially fixed. The node generation process is interrupted in the presence of constraints in order to mitigate the effect of discretization and to generate nodes at relevant locations of the search space (see Fig. 4.3). Since the search space is generated incrementally, the location of each node depends on the discretization of control variables (ESF,γ_T,δ_{ab},Conf) and the independent variable s. Therefore, the objective is to obtain a deterministic trajectory for any choice of discretization, so that the impact of discretization is on the solution is limited.



(a) Altitude constraint case.

(b) Maximum speed case.

Figure 4.3: Node generation in the presence of constraints.

Next nodes (also known as children) are developed from the current node (alternatively called parent) with a set of functions that check whether the trajectory linking both nodes - an edge of the graph - crosses any constraint or state boundary. Figure 4.3(a) shows

how the new node is recalculated at the constraint distance; the intersection between the segment and the vertical line defined by the constraint yields ds and h_{i+1}. With that information, V_{i+1}, m_{i+1} and t_{i+1} are computed by the equations of motion. The process is analogue in the presence of a speed constraint.

Similarly, minimum and maximum speeds are checked to ensure compliance with the flight envelope as displayed in Fig. 4.3(b). In this case, the next node falls out of the search space ($V_{i+1} > V_{max}$) due to the ds value. The algorithm computes the intersection between the edge linking both nodes and the maximum allowable speed (V_{max}), which gives the new ds and a node whose speed is V_{max}. Then, h_{i+1}, m_{i+1} and t_{i+1} values are computed from the new information. It is noted that speed checks are performed in V_{CAS} in order to neglect the effect on true airspeed due to altitude changes. Once all constraint-checks are done and before generating the final node, the values of such states are checked to ensure compliance with the set of constraints, since some speed constraints change as a function of altitude. The A* algorithm continues the node generation and selects at each iteration the node with the least f-cost, so back and forth jumps in the graph may occur, and consequently state boundaries are recomputed constantly. Moreover, depending on control variables discretization, several nodes may be generated very close one from another. These nodes are essentially considered the same [98] if they fall within a neighborhood zone described by the ellipsoid obtained from distance, altitude and speed thresholds, as it is displayed in Fig. 4.4. The threshold values can be modified depending on the flight phase or position in the graph.

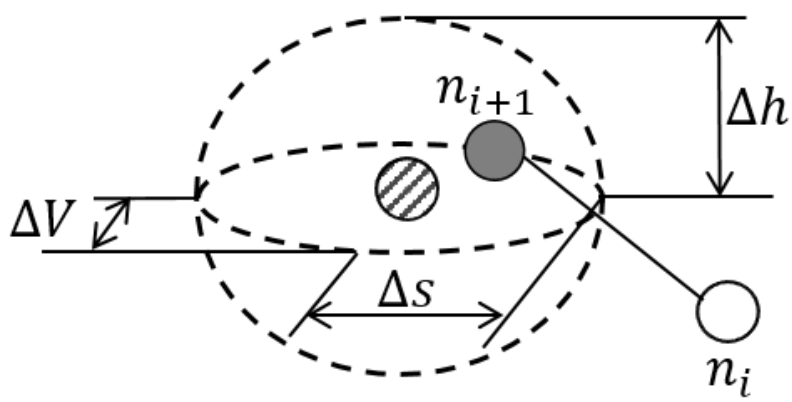
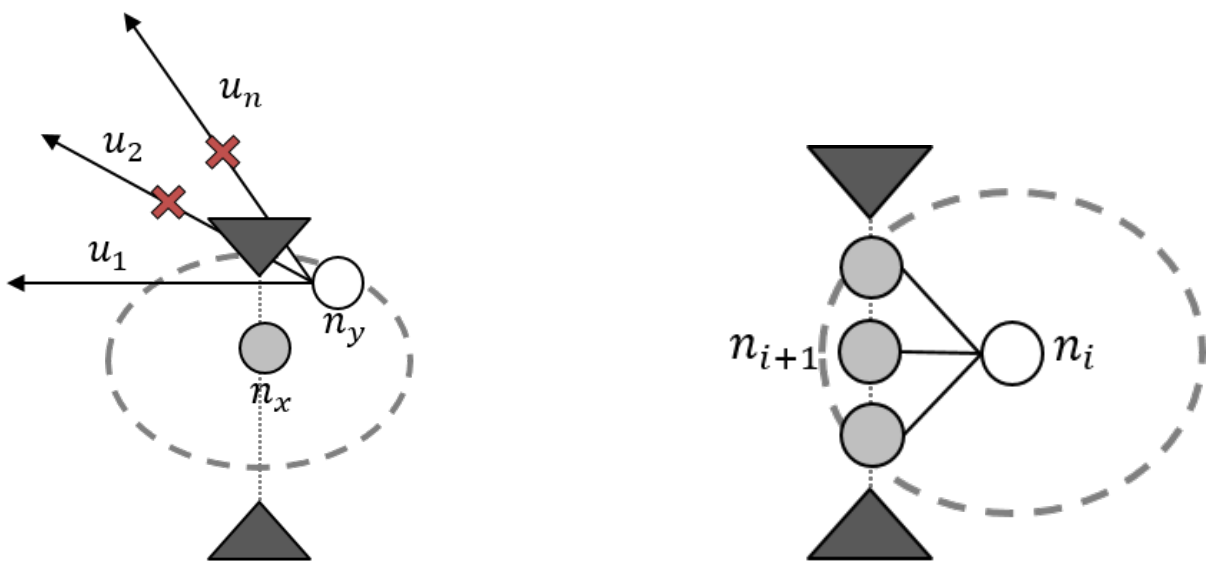


Figure 4.4: Neighborhood zone defined around an already existing node (gray-stripped node), which is at the center of the ellipsoid. The next node n_{i+1}, generated from the current node n_i, falls in the zone.

This simplification implies that any node n_{i+1} being a child node of n_i and dropping in the neighborhood zone of a node already stored (gray-stripped node in Fig. 4.4) is equivalent to that node. On one hand, if this node is stored in the open list, then g-costs

are compared and only the less costly node is stored. On the other hand, if the node is in the closed list, and assuming that the heuristic function is consistent, there is no need to revisit the node and consequently n_{i+1} is discarded. The selection of ΔV, Δh and Δs is challenging, since small values usually lead to large computation times whereas large values may yield unacceptable solutions. In the presence of an altitude constraint, a large neighborhood zone could potentially remove promising candidates as shown in Fig. 4.5(a). In that case, node n_y is generated at iteration y and falls within the neighborhood zone of node n_x, already stored in the open list and generated at previous iteration x. It is assumed that node n_y has a lower g-cost than n_x so that the state is updated in the open list. The distance of n_y is slightly different to that of n_x, and now n_y is not place at the same distance as the constraint. If the distance threshold is too large, then the node may be relatively far from the altitude constraint and most controls $u_2 ..., u_n$ will be rejected, since the next node falls out of boundaries. The only valid control is u_1, which generates a level-off to pass the constraint. This situation generates a dead-zone. It can be deducted that, with a smaller neighborhood area, the primary node n_x would have not been updated. This example illustrates the importance of a well-defined neighborhood area, especially in the vicinity of constraints.



(a) Rejected control policy u_n from the updated node n_y.

(b) Children nodes n_{i+1} within the neighborhood area of their parent node n_i.

Figure 4.5: Issues associated to the neighborhood area in the vicinity of a constraint.

In a similar way, the design shall deactivate the altitude constraint check when the distance between the parent node and the constraint is lower than the neighborhood zone [98]. Otherwise, it causes a dead-zone, since next nodes n_{i+1} of the current node n_i will be intersected at the altitude constraint, but they could not be generated as they fall within the neighborhood area of their own parent node n_i. This situation is illustrated in

Fig. 4.5(b), where the three nodes generated are discarded since the algorithm considers that they are within the neighborhood of an already explored node, which is their parent node n_i already in the closed list. Similarly, a convergence zone is defined around the target node to ensure the convergence of the algorithm. Basically, any node falling in this zone is considered as the target. The design of the convergence zone shall be analogue to that of the neighborhood area; the values shall be equal or lower than those defined for the neighborhood zone, otherwise the algorithm may not find a solution. This area is introduced in the design of the algorithm because it is unlikely that any combination of control variables generate a final state that coincides exactly with the target node. Additionally, this fact justifies the reason why the computation is performed upstream; arriving close to the aircraft current position is important but any bias can be compensated with a proper guidance law. On the contrary, it is mandatory, for obvious safety reasons, the aircraft meets accurately the stabilization point at the proper energy state.

4.5.1 Design of the approach phase

The approach phase requires a different design than the descent part. In a state-of-the-art FMS, the transition of phase is performed as soon as the aircraft starts the deceleration to the approach speed. This phase transition is displayed on the ND through the deceleration pseudo-waypoint.

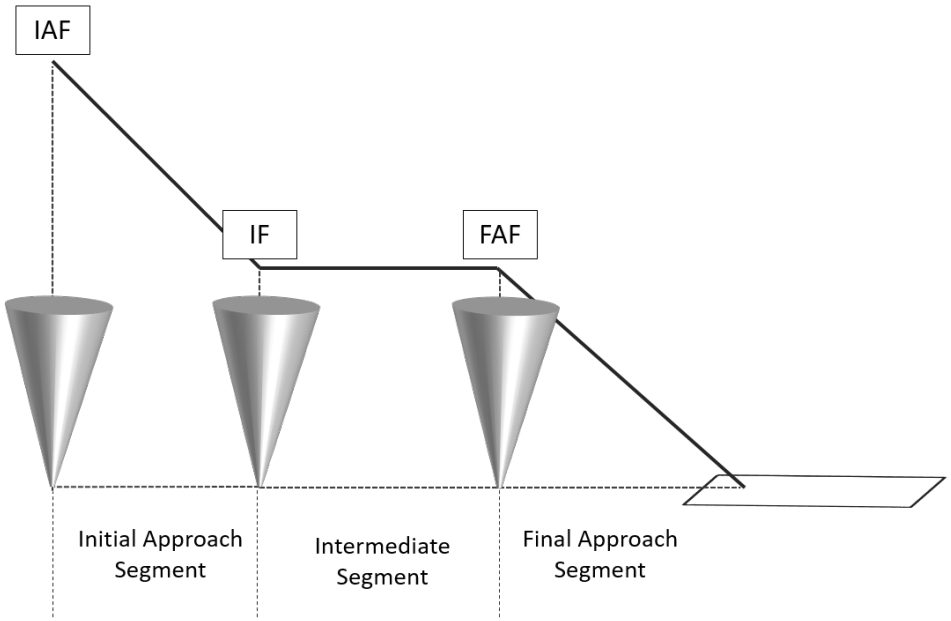


Figure 4.6: Several strategies for flight path construction on final approach.

In this thesis, the transition from descent to approach phase has to be redefined, since the aircraft airspeed varies continuously. Therefore, the approach phase starts as soon as

the aircraft is not in clean flap configuration. This phase is composed of three segments: an IAS, an IS and a FAS. Traditionally, aircraft follow a -3 degree flight path from the FAF until the runway threshold. Thus, most precision approaches are coded with a final slope of around $-3°$. During the final approach, pilots usually set 3 and full flap configurations as the aircraft decelerates towards the approach speed. This type of operation is usually referred as decelerated approach. The aerodynamic characteristics are different from one aircraft model to the other, and similarly the deceleration capability changes. In order to generalize the design of the node generation in approach and optimize with respect to FMS design, the number of combinations have been enlarged.

In Fig. 4.7, three strategies are depicted, which are considered in the A* node generation process during approach phase:

- u_1: aircraft accelerate (upstream from stabilization point) at idle thrust maintaining a -3 degree slope.
- u_2: aircraft acceleration too slow or even decelerates at -3 degree slop, which may occur depending on the aircraft model and flap setting. In this case, speed and slope are compensated through engine thrust.
- u_3: in this strategy, aircraft maintain current speed and flap configuration without following the -3 degree slope but a steeper path.

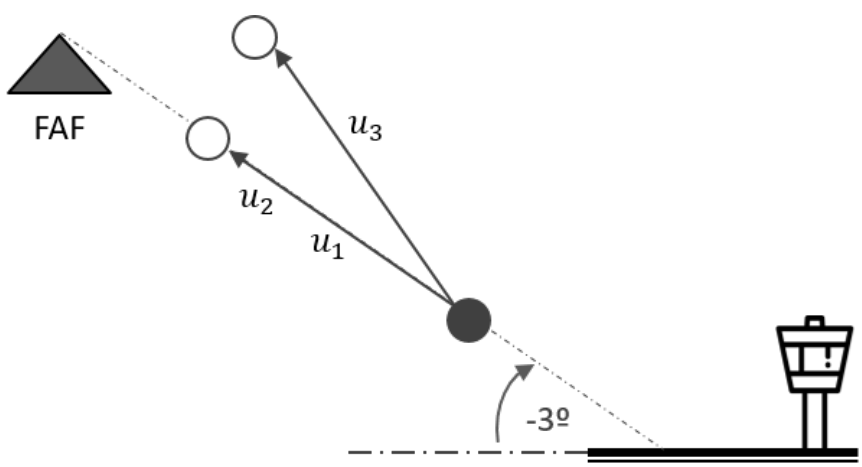


Figure 4.7: Several strategies for flight path construction on final approach: u_1 idle thrust on glide, u_2 constant speed on glide and u_3 constant speed idle thrust.

In strategy u_1 and u_3, thrust is set to idle whereas u_2 requires extra thrust. The former strategy, u_2, is only considered during final approach as it may be necessary to satisfy an "AT" constraint.

4.6 Data structures of open and closed lists

The design of an open ($\mathcal{O}$) and closed ($\mathcal{C}$) list is relevant to assess the performance of the algorithm. Basically, A* performs repetitively three types of operations:

- **Membership:** operation that checks whether the node is already in the open list.
- **Insertion:** operation that inserts the node in the open list.
- **Best-node removal:** operation that finds and removes the best node from the list.

There is another operation that combines several of the previously presented processes. This operation is called increase-priority and occurs when a new node is already in the open list and has a lower g-cost than the already stored node. In that case, the whole list is explored (membership), the existing node, whose cost is higher than the new node, is removed (best-node removal) and the node is finally inserted (insertion). Table 4.1 compares the cost of each operation represented in big-O notation according to different data structure implementations. Basically, this notation classifies algorithms by relating their performance as the input size growths [99]. The comparison is based on the maximum size of the open list, also known as fringe ($\mathcal{F}$) [77].

Data Structure	Membership	Insertion	Best-node
Unsorted	$O(\mathcal{F})$	$O(1)$	$O(\mathcal{F})$
Priority	$O(\log \mathcal{F})$	$O(\mathcal{F})$	$O(1)$
Binary heap	$O(\mathcal{F})$	$O(\log \mathcal{F})$	$O(\log \mathcal{F})$
Hash table	$O(1)$	$O(1)$	$O(\mathcal{F})$

Table 4.1: Comparison of operation costs for typical data structures.

Big-O notation defines the asymptotic behavior, so the choice depends on the size of the fringe and the implementation itself. Since the algorithm presented in this paper generates nodes incrementally, the heuristic function becomes crucial as the more accurate it is, the fewer nodes will be expanded. The closed list collects all expanded nodes and, in that case, only membership and integration operations are performed. In general, A* open list is implemented through a priority list where nodes are sorted in descending order from the most promising to the least. The first node is retrieved at each iteration and the list is reordered with its children nodes. However, the A* version presented in this thesis implements a simpler approach; all nodes are contained in an unsorted open list, the membership operation is defined by a boolean and at each iteration the whole list is explored to identify the less costly node. This choice results from an analysis performed with a simplified version of the A*, which highlighted a higher efficiency of unsorted lists instead

of priority queues for a moderate number of nodes in the fringe (e.g. 2000 nodes). Binary heap and hash tables rely on indexing techniques to remove the best node but membership operation results expensive. In this sense, an hybrid implementation benefits from the advantages of each technique potentially providing higher performance. Nonetheless, some of these solutions are not implementable as the search space is increasingly generated and software architecture optimization remains out of the perimeter of this thesis.

4.7 Control variables discretization

The discretization of continuous state variables determines the performance of the algorithm and quality of the solution. Short step sizes usually provide better results but require lots of computations and contribute to the curse of dimensionality issue. Therefore, the step-size for each continuous control variable should be adjusted with care.

$$u(s) = \{ESF, \delta_{ab}, \gamma_T, \text{Conf}\} \tag{4.1}$$

The control variables ESF, δab and Conf are independent variables whereas the value of γ_T is dependent on the three former control variables. Furthermore, Conf variable is only applicable to approach phase when the aircraft is no longer in clean configuration and flap changes shall occur. The control variables presented in this thesis are adaptive in the sense that they depend on the current flight phase and aircraft condition. Basically, the algorithm implements a certain logic to check that the control value is applicable, otherwise it discards the value without the need of developing the next node. The choice of the control variables ESF and γ_T has been done because they provide an energy representation of the aircraft. The γ_T represents the energy budget whilst ESF defines the energy modulation, which represents the distribution of the budget between the kinetic and potential energy. Moreover, this formulation allows to correlate both control variables to the aircraft altitude and speed. The description of each of the control variables is provided in the following subsections as well as the justification of the discretization choice.

4.7.1 Energy share factor (ESF)

The first control variable to analyze is the ESF, which represents the percentage of the total energy rate dedicated to decelerate the aircraft. The influence of positive energy share increments on the trajectory construction is given in Fig. 4.8:

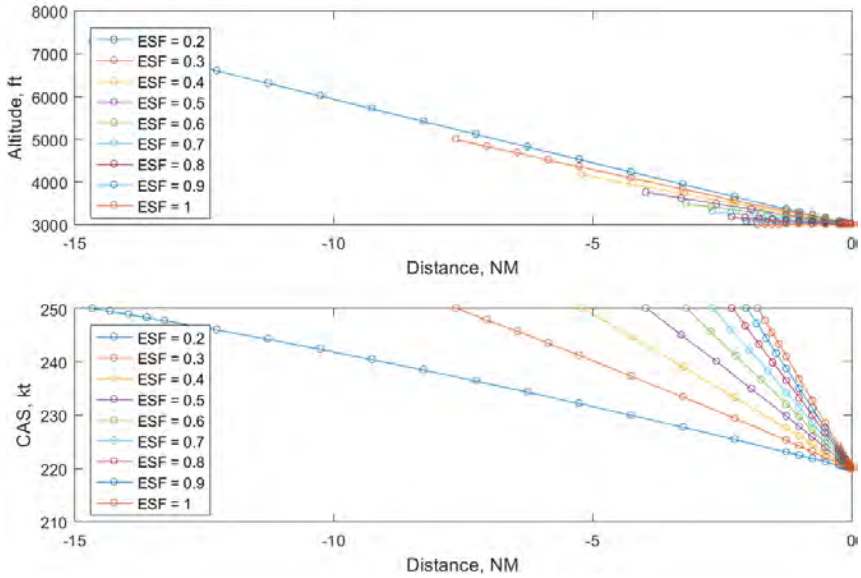


Figure 4.8: Impact of 10% energy share increments on altitude and speed for $ESF > 0.1$. The speed target is 250 knots; small values of ESF lead to long distances while higher values yield shorter distances.

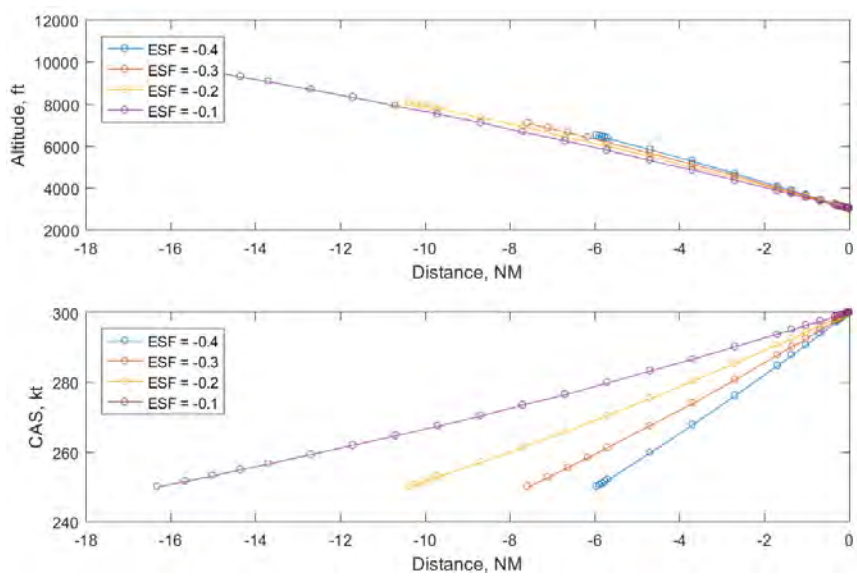


Figure 4.9: Impact of 10% energy share increments on altitude and speed for $ESF < 0$. The speed target is 300 knots; small negative values of ESF lead to long distances while higher values yield shorter distances.

It is observed from Fig. 4.8 that an ESF equal to 1 decelerates the aircraft in 2 NM whereas the same deceleration for a value of 0.2 is achieved in 15 NM, under identical meteorological conditions. The same analysis can be performed for negative ESF where aircraft converts part of the potential energy loss into kinetic energy. In Fig. 4.9), the ESF

ranges from -0.4 to -0.1 and the aircraft accelerates from 250 kt to 300 kt in the forward direction. From the previous figures, it can be concluded that increments of 10% are sufficient for the generation of nodes. Small increments do not represent a real difference and would generate too many nodes, largely increasing the computation time. Moreover, the lowest value has been limited to -0.4 in order to avoid too steep segments, so that:

$$ESF = \{-0.4, -0.3, ..., 1\} \tag{4.2}$$

4.7.2 Airbrakes extension

The second control variable to analyze is airbrakes deflection and its effect on drag. Airbrakes lever on Airbus aircraft set three detent positions as displayed in Fig. 4.10(a): zero, half and full airbrakes. However the pilot can command any other configuration out of the three detents (continuous function). The flight crew order is sent to flight control computers that command through the actuators a certain deflection angle in function of the flight condition. On the avionics side, FMS only considers zero or half airbrakes positions for the construction of the trajectory. On the contrary, the discretization proposed in this algorithm considers three positions (zero, half and full airbrakes) in order to be consistent with current flight controls. Additional values could be considered. In general, full airbrakes shall be avoided as they produce vibrations and structural fatigue but the extra drag may solve discontinuities in the vertical path. Since airbrakes increase the stall speed V_{LS}, their utilization shall be limited. For doing so, the algorithm does not consider airbrakes extension in approach phase except for high-energy scenarios, where their extension is mandatory to increase energy dissipation. This design choice avoids the systematic construction of airbrake segments, and prioritizes other type of segments even when the approach path is stretched.

$$\delta_{ab} = \{0, \delta_{ab_{1/2}}, \delta_{ab_{Full}}\} \tag{4.3}$$

(a) Airbrakes lever.

(b) Flap lever.

Figure 4.10: Airbrakes and flap control lever on Airbus aircraft.

4.7.3 Total flight path angle (γ_T)

The third variable is the total flight path angle, whose value depends on thrust setting and drag. Fuel consumption is minimized when engines are idle but time is usually saved at higher speeds, which may require to add thrust in order to keep airspeed high. In-between there are plenty of combinations that could be considered. For simplification purposes, only three thrust positions are considered. The first thrust value corresponds to the minimum idle position ($\gamma_{T_{min}}$). The second value is equal to the thrust required to maintain a level-flight at constant CAS speed ($\gamma_T|_{T=D}$). In that case, γ_T is zero as energy is neither dissipated nor aggregated (thrust compensates drag, $Thr = D$). The third thrust value is only applied to time optimization ($\gamma_{T_{max}}$), and corresponds to climb thrust (80% of maximum thrust). Furthermore, idle thrust can be combined with airbrakes extension in order to increase energy dissipation ($\gamma_{T_{Full}}$ and $\gamma_{T_{1/2}}$). From the previous assumptions, it can be induced that γ_T values are the result of a combination of thrust and drag. Therefore, γ_T is computed for each iteration and its value depends on ESF and δ_{ab} control variables in descent, and also to the flap setting (Conf) in approach phase. It results in the following combination of γ_T values:

$$\gamma_T = \{\gamma_{T_{Full}}, \gamma_{T_{1/2}}, \gamma_{T_{min}}, \gamma_T|_{T=D}, \gamma_{T_{max}}\} \tag{4.4}$$

4.7.4 Flap configuration changes

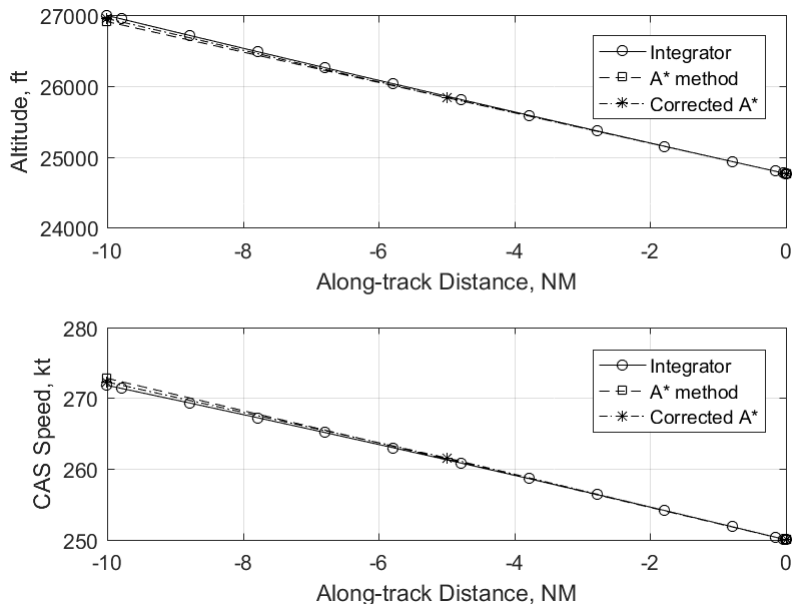
Finally, the forth control variable corresponds to flap configuration, which is commanded by the flight crew through the flap level showed in Fig. 4.10(b). Pilots manually change the aerodynamic performance according to their criterion. The effect of flap setting is an increase of lift coefficient as a result of wing surface extension. The stall speed is decreased and the aircraft generates more drag than in clean configuration. Idle ratings are usually higher to comply with engine certification specifications so the aircraft consumes more fuel. In nominal approach phase, landing gear is extended after Conf-2, however, FMS model considers that the extension is performed at Conf-3 to construct a conservative approach profile.

$$Conf \in \{1, 2, 3, Full\} \tag{4.5}$$

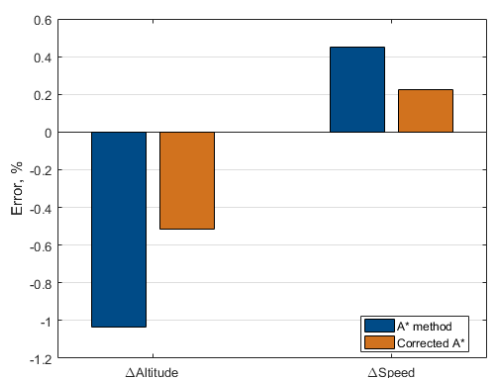
4.7.5 Discretization error quantification and sensitivity analysis

The action of discretizing generally carries a certain error in the computation. The node generation process computes a new node $i + 1$ based on the state of the previous node i as displayed in Fig. 4.11. In practice, it means that the segment between both nodes is

calculated with a fixed thrust, drag, speed and mass (solid line) that comes from node i. After the calculation of the new node a readjustment based on the average value between the state of nodes i and $i+1$ would result in a more accurate estimation (dash-dotted line) but should be done for each generated node and would be prohibitive in terms of computation time. An alternative solution could be to smooth the constructed path in a post-processing phase once the optimal trajectory is found. The size of the error depends on the distance between nodes; the larger the distance step, the bigger the error is.



(a) State error induced by the effect of discretization.



(b) Altitude and speed errors compared with integrators.

Figure 4.11: Impact of discretization in node generation process. Comparison of A* and corrected A* calculations to Runge-Kutta integration.

Figure 4.11 compares three different integrations methods; in the first one, the trajectory is computed with classical integration methods that use intermediate integration points. In the second strategy, the next node is generated from the current node assuming that parameters such as drag, thrust are constant. This is not true as these parameters

change as soon as the aircraft ascends (in the backwards sense). In order to mitigate this error, a corrected method may be applied. Discretization errors become relevant as soon as the distance step is greater than 5 NM. Thus, any segment longer than 5 NM uses an intermediate point that minimizes the error as shown in Fig.4.11. In the given case, the trajectory provides a -0.32% altitude error and 0.36% speed error. While the individual error is not relevant itself, the cumulative error may generate a biased trajectory. The implementation of an intermediate integration point reduces the error by 50%.

4.8 The heuristic function

4.8.1 Properties of the heuristic

The time-complexity of A* algorithm mainly depends on the implementation of the heuristic function. As mentioned in chapter 2, this function is an estimation of the optimal cost and has to satisfy some properties in order to ensure global optimality. The first property is admissibility, which means that the heuristic shall underestimate the actual optimal cost for each node of the graph, which is defined in Eq.(4.6). That is why it is considered as a lower bound. In the particular case where the heuristic equals to the actual optimal cost, A* algorithm develops very few nodes however, in practice, it seldom occurs as it would indicate that the solution is already known a priori.

$$h(n) \leq h^*(n), \qquad \forall n \tag{4.6}$$

Where $h^*(n)$ is the shortest path from any node n to the goal state. Thus, an admissible heuristic shall comply with the following condition:

$$h(n_{tgt}) = 0 \tag{4.7}$$

Where n_{tgt} is the target node. It can be deducted from Eq.(4.7) that the heuristic value at the target node equals to zero. The second property is consistency and indicates that the heuristic function is monotonically decreasing along the path. In that case, a closed list containing all developed nodes can be implemented, and has the advantage of preventing the algorithm from re-visiting nodes already explored, reducing time and space complexity. This property is easy to demonstrate as the only condition is to check that heuristic values decrease as the algorithm gets closer to the target node. Therefore, A* algorithm is optimal if the heuristic is admissible and finds the solution, if one exists (completeness of the algorithm). In reality, for a given h, no other search algorithm will develop fewer nodes than A* in the search of the optimal path (optimally efficient).

Consistency property means that the function is monotone and it can be demonstrated through the triangle inequality, which is displayed in Fig. 4.12.

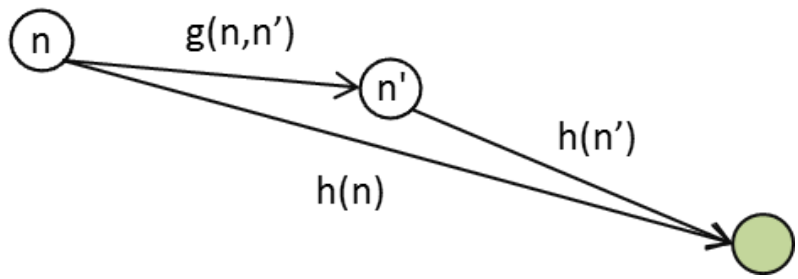


Figure 4.12: Triangle inequality property of consistent heuristics.

Where n is the frontier node, n' is a children node of n and the target node is represented in green. The triangle inequality states that for any pair of nodes (n, n'), where n' is the successor of n, the estimated cost of reaching the target node from n, $h(n)$, shall be less or equal than the cost of reaching n', which is $g(n, n')$, plus the estimated cost from n' to the target node ($h(n')$):

$$h(n) \leq g(n, n') + h(n') \tag{4.8}$$

4.8.2 Definition of the heuristic for trajectory optimization problems

This section aims to describe the different heuristic functions that have been implemented and analyzed in the A* version presented in this thesis. The assessment technique to check admissibility and consistency properties is the following; on one hand, admissibility is checked by retrieving the nodes of the optimal path. For each node, the value of the heuristic is compared to that of an ideal heuristic, which is the difference between the actual optimal cost at the end of the computation and the cost-to-go to the current node. The ideal heuristic is also called the exact heuristic for each of the node, since the algorithm would have converged immediately. The idea is based on the fundamental hypothesis that, if all nodes belonging to the optimal path are admissible, there is no reason to think that the rest will not be. On the other hand, consistency property is checked by plotting the heuristic estimation of each generated node as a function of distance, and then check that the function is monotonically decreasing.

A flight performance study [100] demonstrated that, for a given CAS speed, the straight trajectory between two points (geometric segment) is less efficient in terms of fuel and time than a level-off followed by an idle segment. This is observed in Fig. 4.13 where, for 10.000 feet altitude change to be done in 40 nautical miles, the geometric segment (dashed line) consumes more fuel and time than a level-off followed by an idle segment (solid line),

as it is shown in Fig. 4.14. The use of full airbrakes extension (dash-dotted line) does not provide any advantage but penalizes fuel consumption as the idle distance traveled is shorter than without additional drag. In general, level-off segments require more thrust but for a shorter distance than geometric segments that require thrust all along the path. Time is reduced for the same CAS speed, since the speed relative to the air mass (TAS) increases with increasing altitude under the same environmental conditions. It is noted that for confidentiality reasons, no flight performance is displayed in Fig. 4.14.

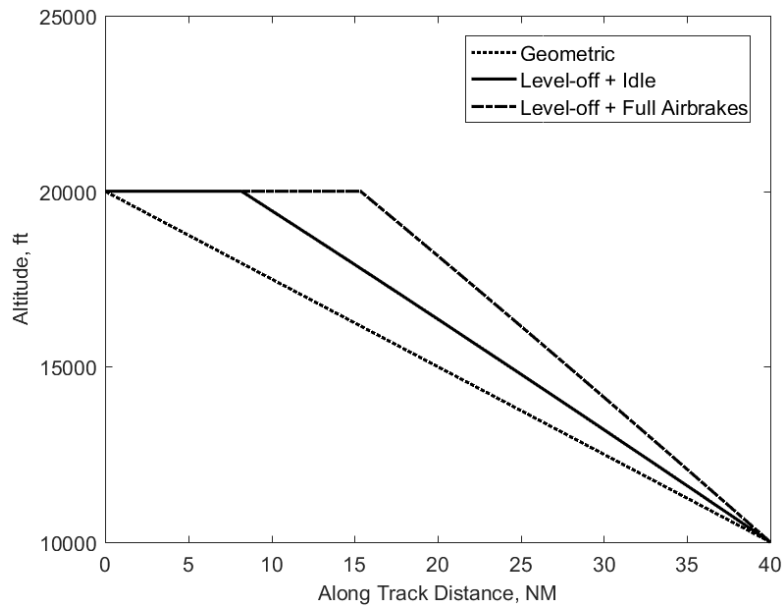


Figure 4.13: Comparison of three type of segments: geometric (dashed line), level-off plus idle (solid line) and level-off plus full airbrakes extension (dotted-dashed line).

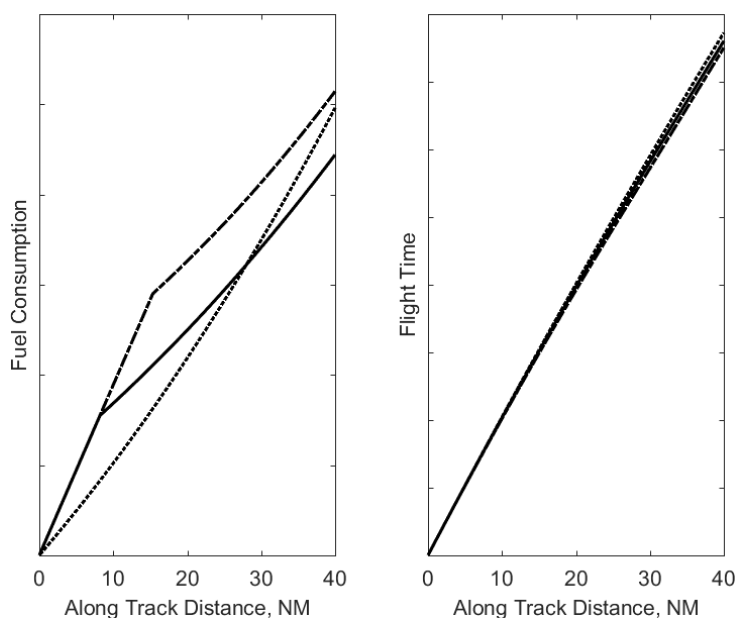


Figure 4.14: Fuel consumption and time flight performance comparison.

4.8.3 Manhattan-distance based heuristic

The Manhattan distance is a powerful alternative to the euclidean distance - defined by the straight distance between two nodes as shown in Fig. 4.15 - as it is computationally cheaper and reduces the computational effort in large scale problems. Results obtained from the study displayed in Figs. 4.13 and 4.14 suggest that heuristics based on euclidean distances may not be admissible in the vertical plane. In ideal atmospheric conditions, geometric segments consume more fuel than a level-off followed by an idle segment. In the former, thrust is not idle for the whole segment whereas in the latter, extra thrust is only required for the level-off. Similarly for flight time, for a given V_{CAS} speed, true airspeed increases as a function of altitude, so that at higher altitudes flight time decreases. Therefore, Manhattan distances consider longer distances and may be a convenient alternative for the definition of admissible heuristics.

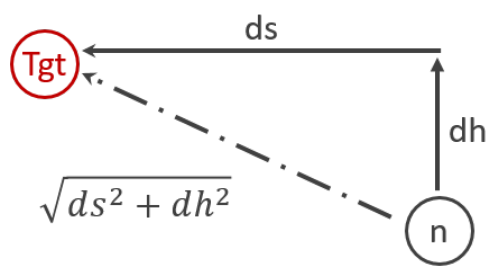


Figure 4.15: Comparison between Manhattan (solid line) and euclidean distance (dash-dotted line)

The cost function of the problem is the sum of fuel and time costs. The fuel burn between a pair of nodes is not easy to estimate without integrating the segment, but this approach for each pair of nodes of the search space would be computationally prohibitive. An alternative method could be to define an average fuel flow rate that would multiply the Manhattan distance. However, the average flow rate has to be defined with care in order to avoid overestimating the optimal cost, thus resulting in a pseudo-optimal solution. A generic and conservative approach is to define as reference fuel flow the lowest value, which is the one from the target node, since for a given speed and thrust rating, the higher altitude the lower the fuel flow. The estimation of time is the manhattan distance divided by the maximum operational speed (V_{MO}). This heuristic function works relatively well for nodes close to the target but is excessively underestimating for the rest of nodes, which leads to long computation times. In general, computations using an heuristic function based on the Manhattan distance gave poor results in terms of computational performance.

4.8.4 Flight Performance Heuristic

The previous heuristic was admissible but did not represent an advantage in terms of computation time. Aircraft performance knowledge combined with a relaxed version of the problem (constraintless) present an opportunity to define a meaningful heuristic function. From a performance perspective, the greendot speed ($V_{CAS_{Gdot}}$) is defined as the best lift-to-drag ratio speed that leads to the longest glide distance. As a consequence, a short approach followed by a continuous descent at $V_{CAS_{Gdot}}$ could be used as a lower bound for the fuel optimization case. The computation of the heuristic is performed at the beginning of the algorithm through an integrated path as displayed in Fig. 4.16, which generates a two dimensional lookup table. The figures displayed hereinafter have been computed using BADA performance model.

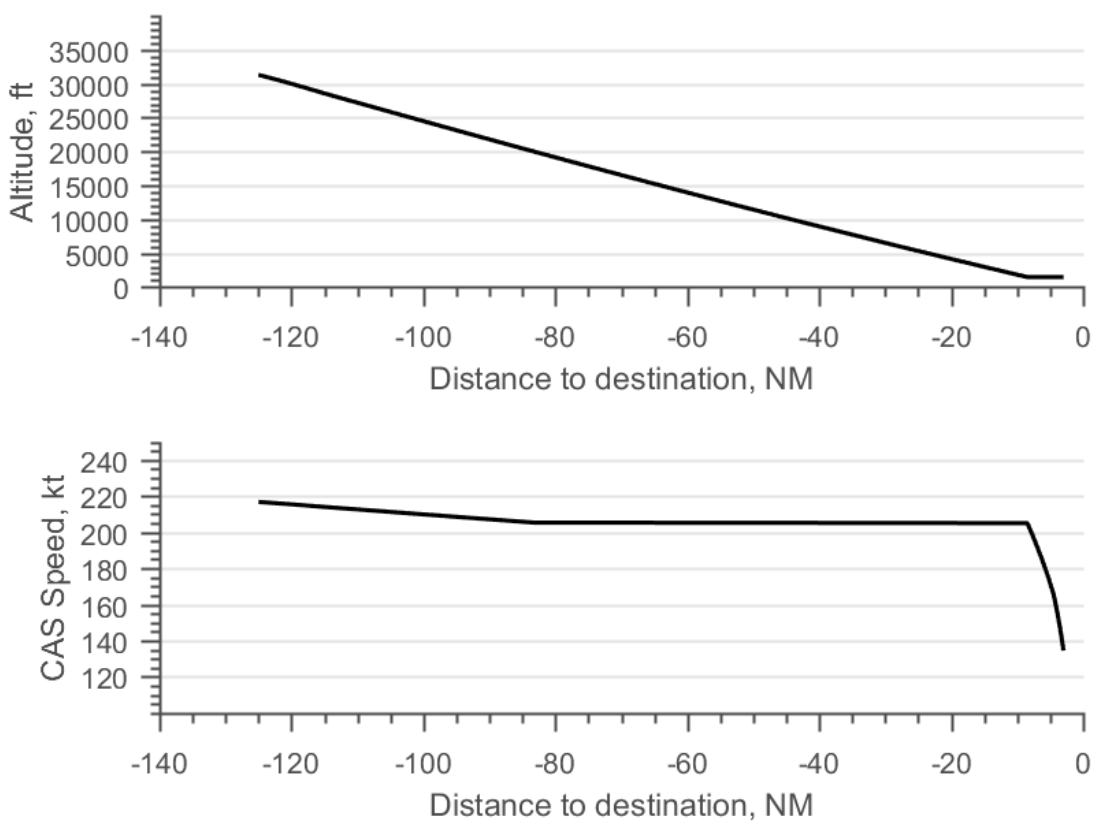


Figure 4.16: Flight performance heuristic using BADA: computation of altitude and speed profiles. For each node of the graph, the heuristic estimation is the cost from the current node to the target node. Linear interpolation to obtain this value.

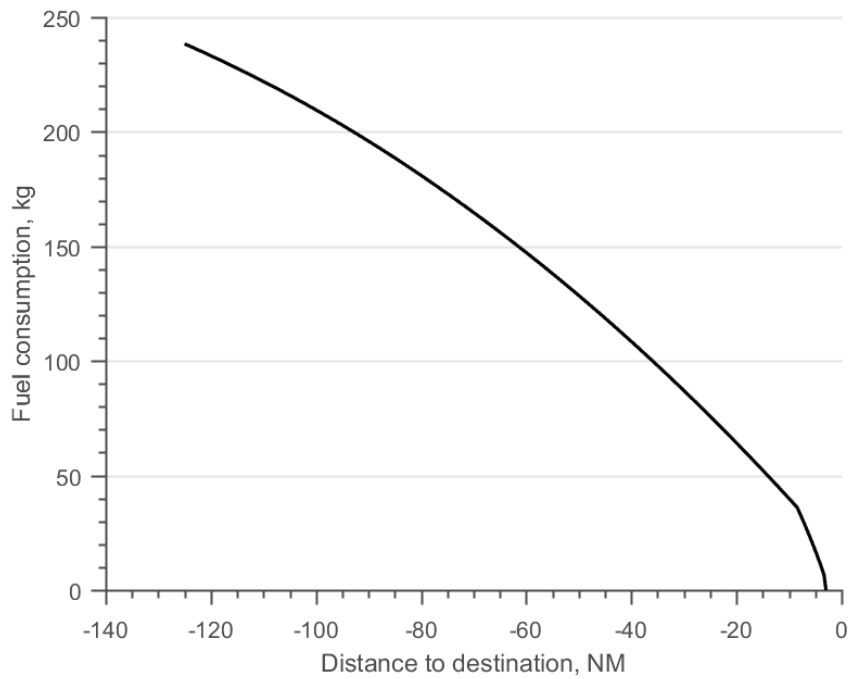


Figure 4.17: Fuel evolution of flight performance heuristic using BADA.

For each generated node, the heuristic function is obtained through the mass difference between the target node and the interpolated mass at the current node distance, which is obtained from Fig. 4.17. The disadvantage of this heuristic function is that all nodes located at the same distance produce the same estimation, so node discrimination is done based on $g(n)$ cost.

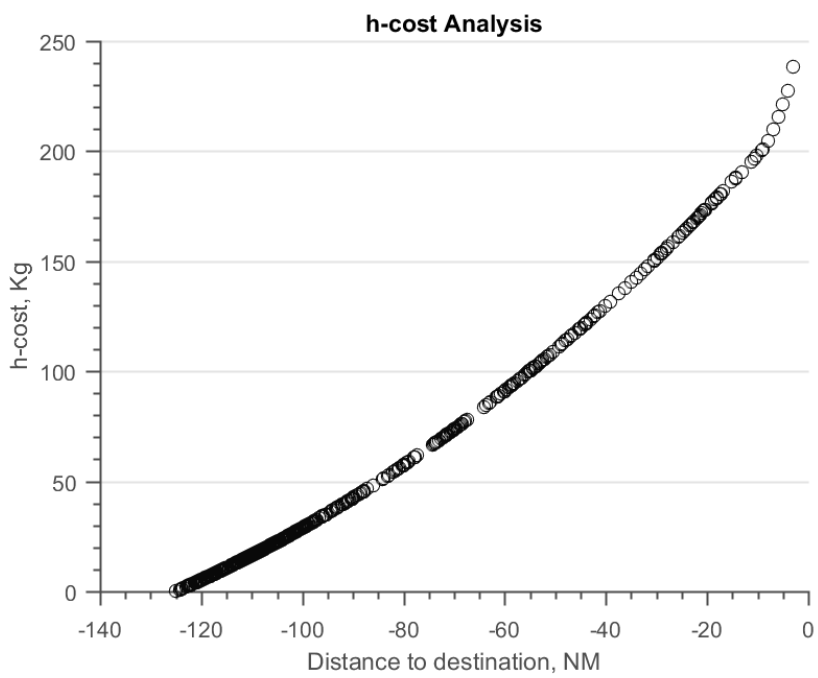


Figure 4.18: Monotononically decreasing heuristic function.

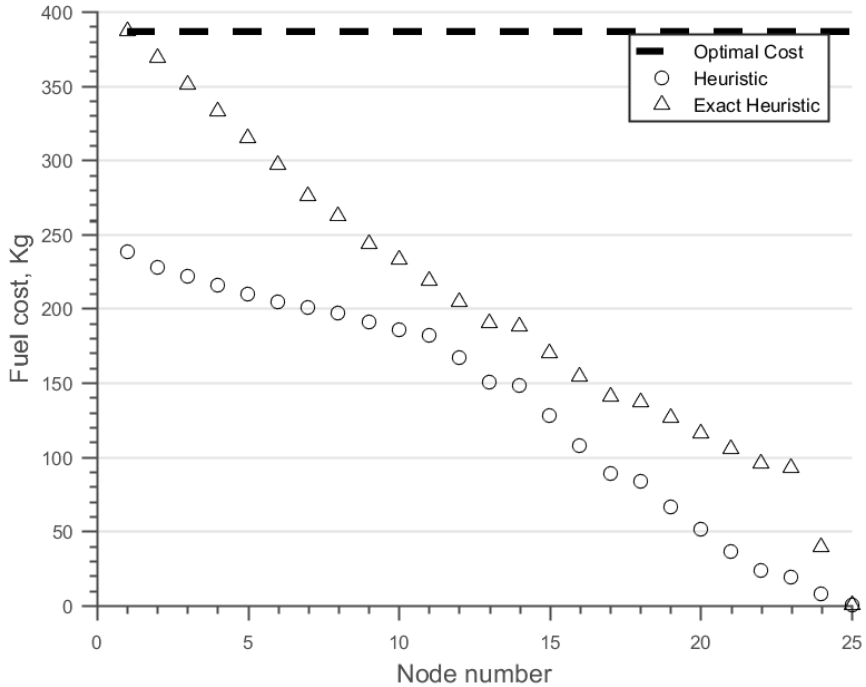


Figure 4.19: Admissibility criterion of the heuristic function.

The heuristic has proved to be consistent as the function is monotonically decreasing as displayed in Fig 4.18. In addition, the admissibility criteria is respected for all nodes belonging to the optimal path as their estimation is optimistic with regards to the actual optimal cost as seen in Fig. 4.19. The exact heuristic represents the difference between the actual optimal cost and the node cost g. Note that the heuristic for each node belonging to the optimal path underestimates the actual optimal cost.

4.9 Conclusion

On one hand, the methodology proposed in this thesis generates the optimal trajectory for any existing arrival procedure at any airport in the world. This generalization enlarges the number of possibilities to look at because what works for one procedure may not work for another. This is the first complexity that has been overcome during the thesis. Furthermore, A* methodology easily implements state and control bounds and, unlike other techniques, the more constraints the problem has, the faster the algorithm computes an optimal path as the search space is pruned. In this sense, the constraint set could be easily enlarged in order to construct a trajectory operationally similar to those flown by pilots nowadays. It is a trade-off between optimality and operational representativeness.

On the other hand, the computational performance of the A* algorithm mainly depends

on the definition of the heuristic function. Trajectory optimization problems generate large amount of nodes with similar costs so that traditional heuristic functions used in most pathfinding applications do not identify early enough the best candidates nodes. It is important to define the heuristic function correctly, which must satisfy admissibility and consistency properties to ensure the optimality of the solution. The proposed algorithm does not intend to be optimal in terms of computational performance, since that topic is out of the scope of the thesis.

In conclusion, the algorithm presented is this thesis implements a modified version of the classic A* algorithm, which is adapted to the trajectory optimization problem in descent and approach phases. One important contribution of this work is linked to the fact that the search space is constructed gradually and that the node generation process uses the available information of the problem, such as constraints, to prune and reduce the discretization impact on the optimal solution. Moreover, the algorithm implements some logics to adapt the control variables to be used as a function of the current flight phase and aircraft energy state (adaptive control variables). The next chapter introduces three different scenarios in which the optimal trajectory has been computed using the A* methodology presented in the current chapter.

5

Results and discussions

This chapter presents and analyses three case studies for different arrival procedures and aircraft conditions. The first case study uses BADA model for the computation of the optimal trajectory. Then, the second case study compares the optimal profile computed using Airbus FMS model with the one computed by a real FMS software. The trajectory is flown in an Airbus flight simulator with the aim of assessing the operational representativeness. Finally, the third case study presents a high-energy scenario where two different flight strategies are compared. Results show that relevant fuel savings can be obtained as a consequence of a tailored profile, and highlight the potential of the function for airlines.

5.1 Test parameters

This chapter presents a series of scenarios where the algorithm computes the optimal trajectory that accounts for the current aircraft state, in other words, the trajectory connects the destination runway threshold with the current aircraft position regardless of the present flight phase and guidance modes. Then, the resulting trajectory is compared with that of a standalone Airbus FMS simulator hosted in a computer. The optimization criterion for this study is the fuel consumption. In particular, three case studies are presented: case study in section 5.2 computes the optimal trajectory using the BADA model. In this case, the resulting trajectory cannot be compared with the one of a real FMS due to the disparity between the aircraft models. The second case study in section 5.3 applies a real PDB for the computation of the trajectory. This trajectory is compared with that of a state-of-the-art FMS and assessed in the flight simulator. Finally, the case study presented in section 5.4 analyses a scenario where the aircraft is in high-energy condition and proposes a comparison between two vertical trajectories that solve the over-energy situation with different flight strategies.

The calculations have been performed with a Single Aisle aircraft model (A320). This choice is justified by the fact that A320 is one of the most flown aircraft in the world. Assuming that fuel savings differences between fuel short-haul and long-haul aircraft are very small, in terms of percentage, descent and approach phases represent a higher percentage of the total savings per flight for short-haul aircraft than for long-haul, since cruise is the dominant phase in long-haul flights. Besides, these figures increase quickly when accounting for the daily number of operations and airlines fleet, which usually count with a large number of short-haul aircraft, and thereby resulting in a greater impact on the financial results of airlines. The general parameters of the simulation used for the three case studies are summarized in Table 5.1:

Parameter	
Aircraft type	A320
Cost Index, kg/min	0
Wind, m/s	0
Δ ISA, °C	0

Table 5.1: General simulation parameters for the case studies.

Atmospheric conditions have a large impact on aircraft performance, however, ideal atmospheric conditions have been considered for the three case studies. The idea is to separate weather conditions from the generation of the trajectory as otherwise it may be difficult to interpret the results. Finally, the CI has been set to zero, which means that the optimization focuses on fuel consumption. In general, airlines choose a CI ranging

between 20-40 to avoid low cruising Mach speed, since the value is applicable to the entire flight. This makes airlines to define a variable CI depending on the phase of flight. The impact of CI in descent and approach is generally low with typical delays of no more than 3 minutes, which suggests that optimizing fuel in these phases is generally desirable.

5.2 Case study I: Dallas-Fort Worth arrival

The first case study consists in the calculation of the optimal trajectory using BADA. The selected aircraft model is an A320-231 equipped with IAE V2500 engines. The selected STAR is BOOVE4 at KFDW airport (see Fig. 5.2), the transition procedure is GEEKY and the approach is given by an Instrument Landing System (ILS) procedure to runway 18R (see Fig. 5.1). The initial aircraft state corresponds to the stabilization gate whereas the final (target) state is the aircraft current position. These are summarized in the following Table 5.2:

	Initial state	Final state
Distance to destination, NM	-2.92	-125
Altitude, ft	1125	30000
Speed, kt	129.1	235
Flap setting	Full	Clean

Table 5.2: Case study I: Aircraft initial and final state.

The list of applicable altitude and speed constraints for BOOVE4 procedure with the waypoint identifier is provided in the Table 5.3. Altitude constraints are expressed in Flight Level (FL) above 10000 feet and in feet below that altitude. Speed constraints are indicated in CAS-knots.

Waypoint	Type of CSTR	ALT CSTR	SPD CSTR
GIBLT	WINDOW	FL280-FL250	290
DARRB	WINDOW	FL230-FL200	290
BOOVE	WINDOW	FL190-FL170	280
SHMPP	WINDOW	FL170-FL150	280
CURLE	WINDOW	FL150-FL130	270
MOWWW	WINDOW	FL130-FL110	250
DELMO	AT	FL110	220
LEGRE	AT OR ABOVE	3000	

Table 5.3: List of constraints with waypoint labels for KDFW case.

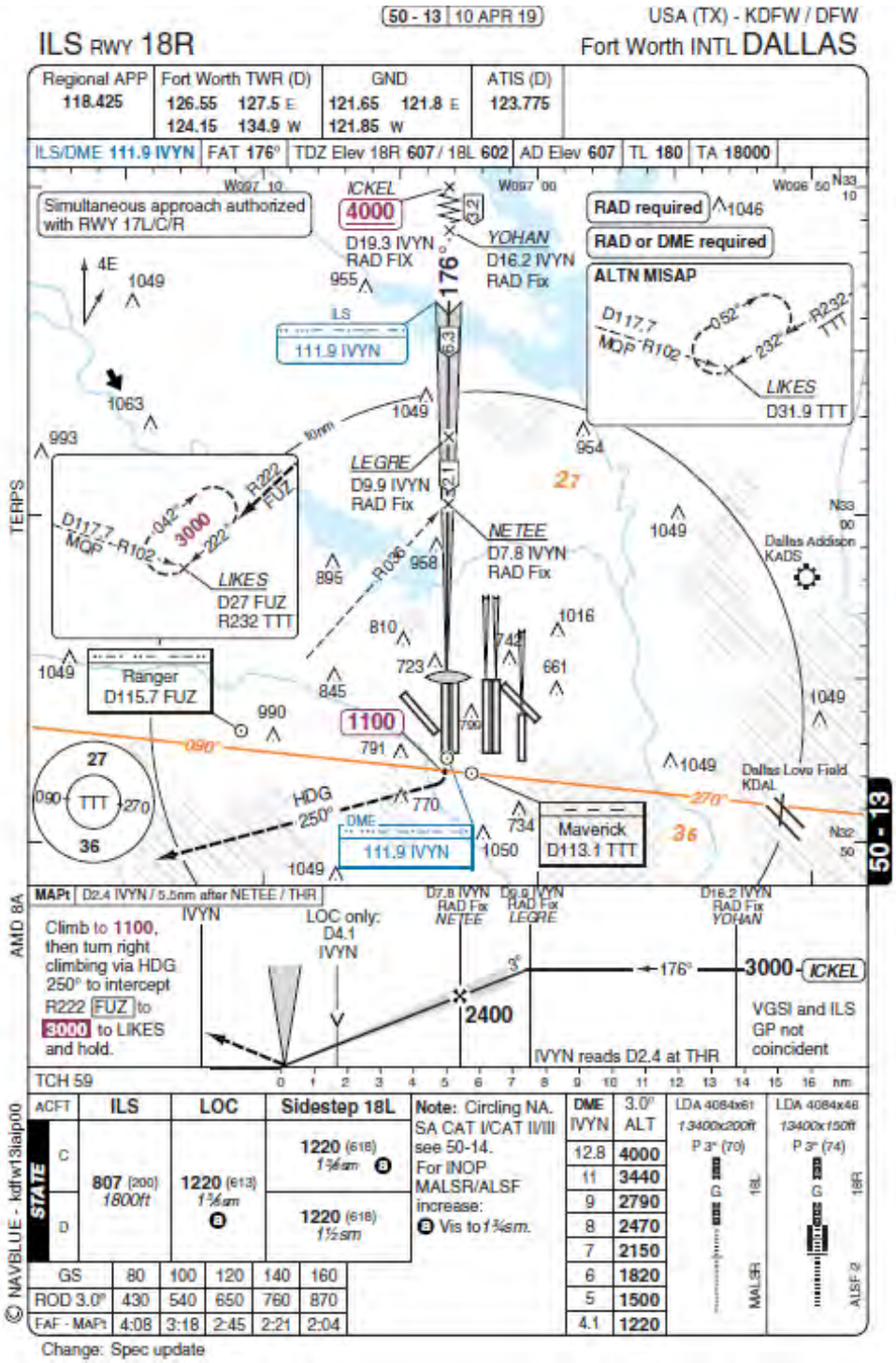


Figure 5.1: APPR ILS-18R. Source: NavBlue.

The approach procedure chart is given in Fig. 5.1 as well as the STAR procedure chart depicted in Fig. 5.2. This type of design in arrival procedures is increasingly popular in the United States, where altitude constraints define a corridor that converge in an AT constraint. The AT OR BELOW speed constraint is operationally interpreted by ATC as an AT, thus the energy state is completely defined. This procedure is particular due

to the fact that DELMO waypoint contains an altitude constraint of 11000 feet combined with a CAS speed constraint of 220 knots, which is quite a low speed for that altitude.

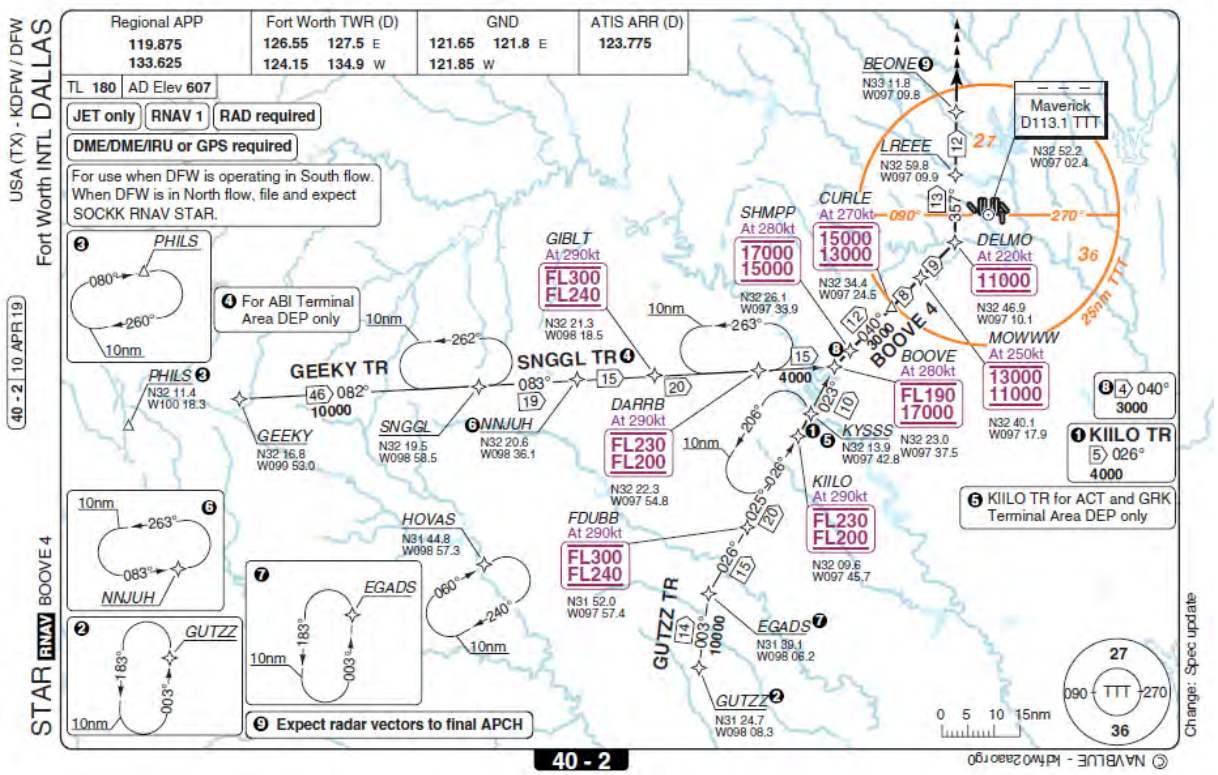


Figure 5.2: BOOVE4 STAR. Source: NavBlue.

5.2.1 Trajectory computation using BADA model

The optimal trajectory for the first case study has been computed using the BADA model [91]. This model contains a large database of different commercial airplanes and also helicopters. It is widely used in the research community as it provides a sufficent level of representativeness for typical aircraft. Nevertheless, the model can not be used for comparison with calculations coming from state-of-the-art FMS, since the aircraft performance data comes from different sources. The resulting altitude and speed profile computed by the A* algorithm for the KDFW arrival procedure is illustrated in Fig 5.3. It is observed that the aircraft continues the cruise phase for several nautical miles. During this period the total energy is maintained through thrust addition, then, the descent is initiated at -118 NM. From the ToD, thrust is set idle while the aircraft flies as high as possible, which is more efficient in terms of fuel, through appropriate energy management.

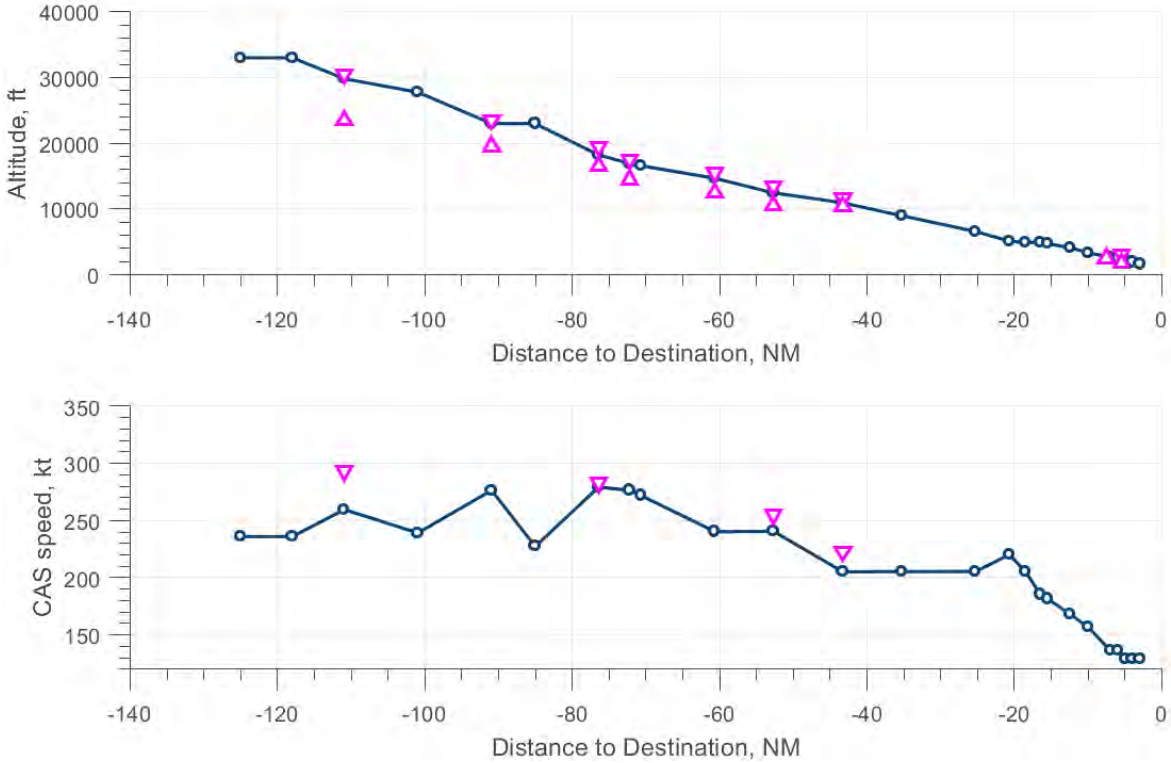


Figure 5.3: KDFW case study: Altitude and speed profile.

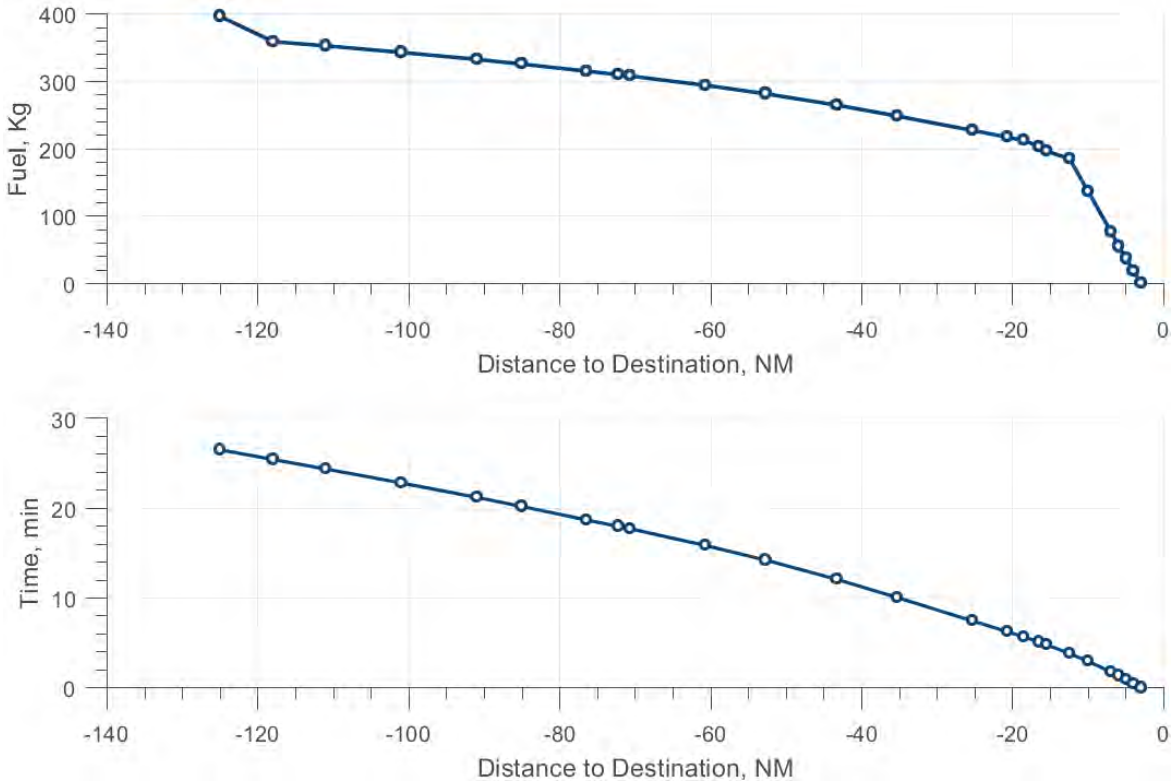


Figure 5.4: KDFW case study: Fuel consumption and flight time.

The speed varies from 215 knots to 270 knots due to the consecutive changes of energy share, as displayed in Fig. 5.5. Altitude and speed constraints are represented by pink-triangles and the aircraft flies as close as possible to their upper values. The trajectory burns 400 kg of fuel with a total arrival time to the stabilization gate of 27 minutes. It is observed from Fig. 5.4 that approximately half of the total fuel is consumed during the approach phase, since flap settings increase the idle rating of the engine to comply with certification regulations.

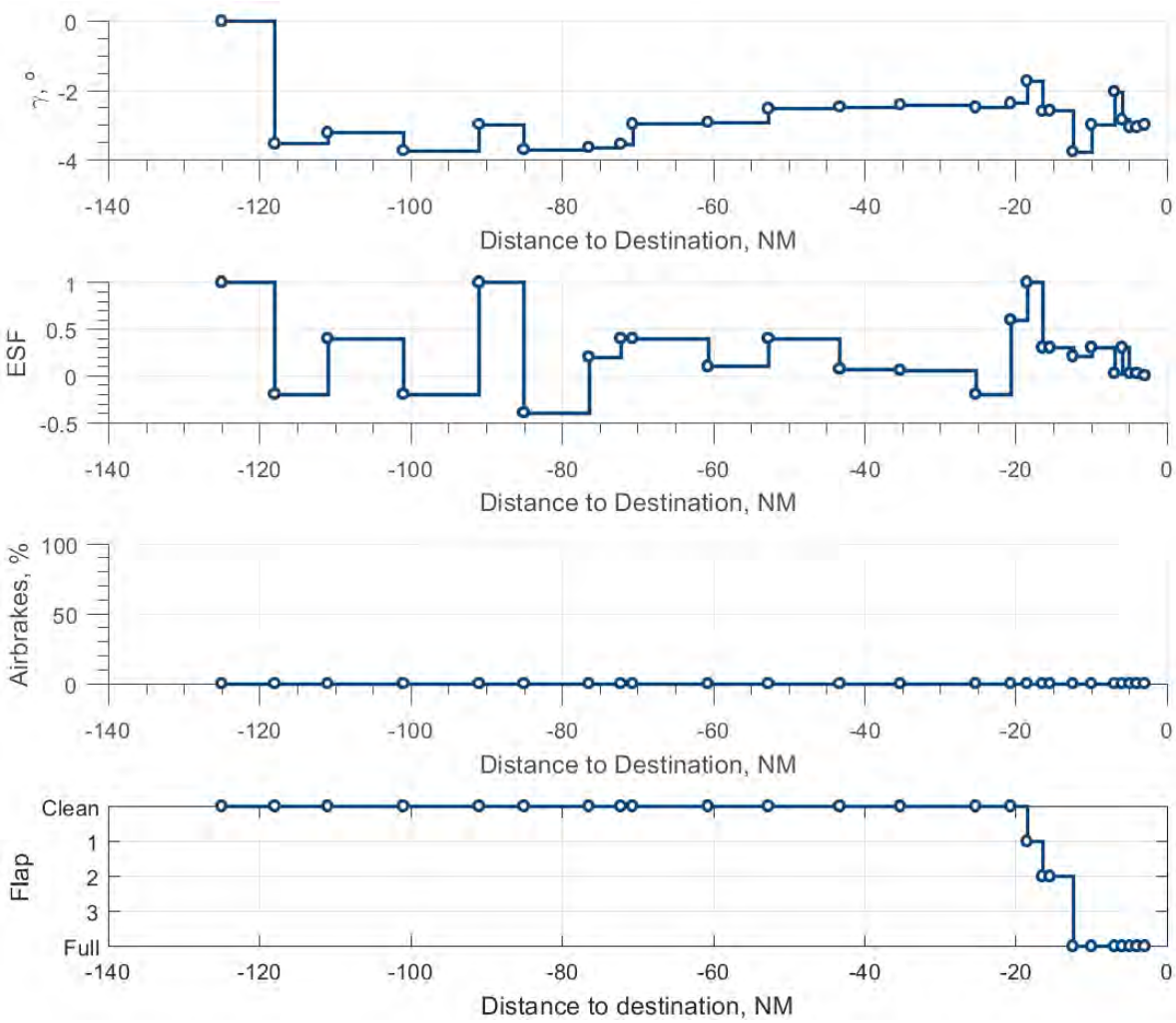


Figure 5.5: KDFW case study: Control variables.

As it can be observed from the previous figures, the aircraft exchanges continuously kinetic energy and potential energy through by the action of the control variables. The energy of the aircraft is restricted by the set of altitude and speed constraints, which impose upper and lower energy bounds. Under this circumstances, the algorithm finds iteratively the most appropriate aircraft energy state with the aim of reducing the fuel consumption of the aircraft. The lateral path of the computed trajectory is given in Fig. 5.6:

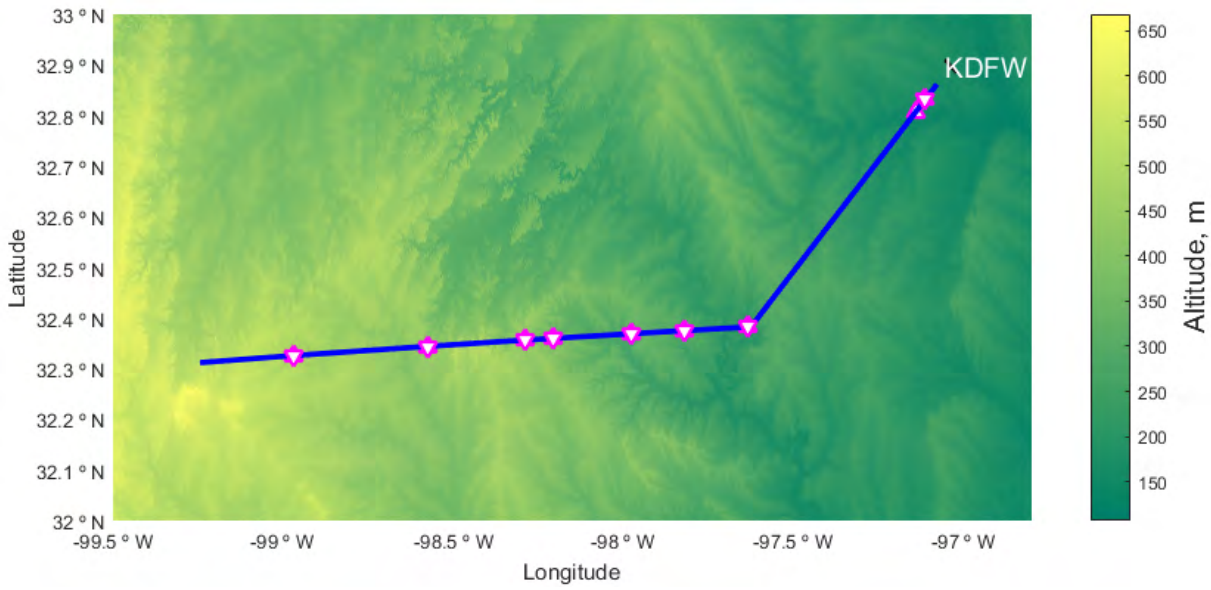


Figure 5.6: KDFW case study: Lateral path.

Then, the three dimensions visualization of the trajectory is provided in Fig. 5.7:

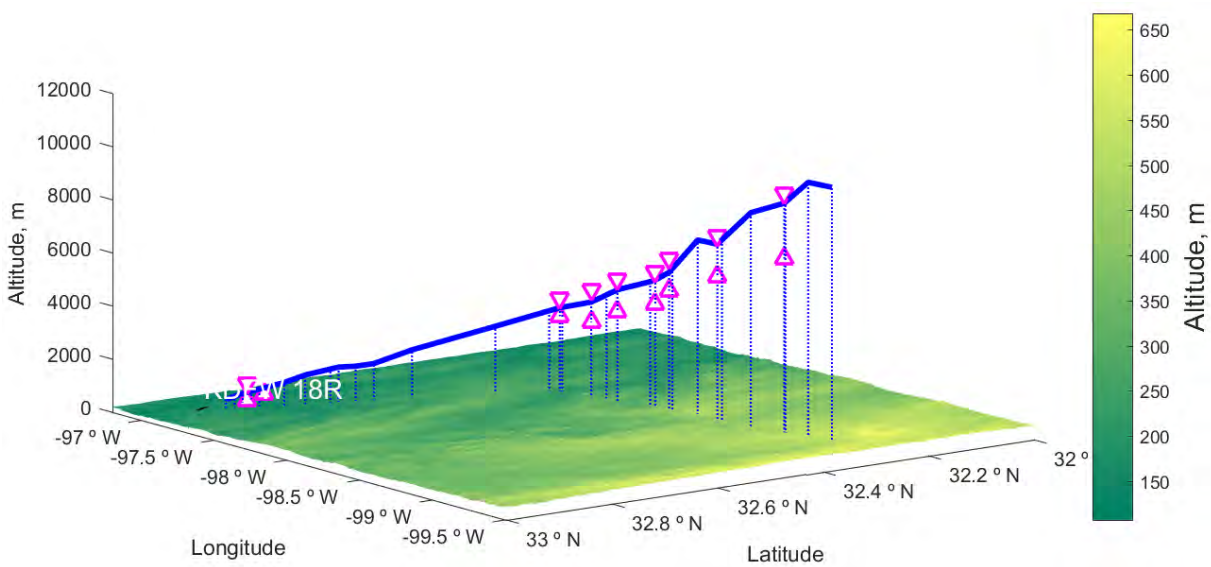


Figure 5.7: KDFW case study: Vertical visualization.

These trajectories have been computed using digital terrain data provided by [101]. The waypoints of the arrival procedure and the integration points used by the algorithm are displayed in Fig. 5.8.

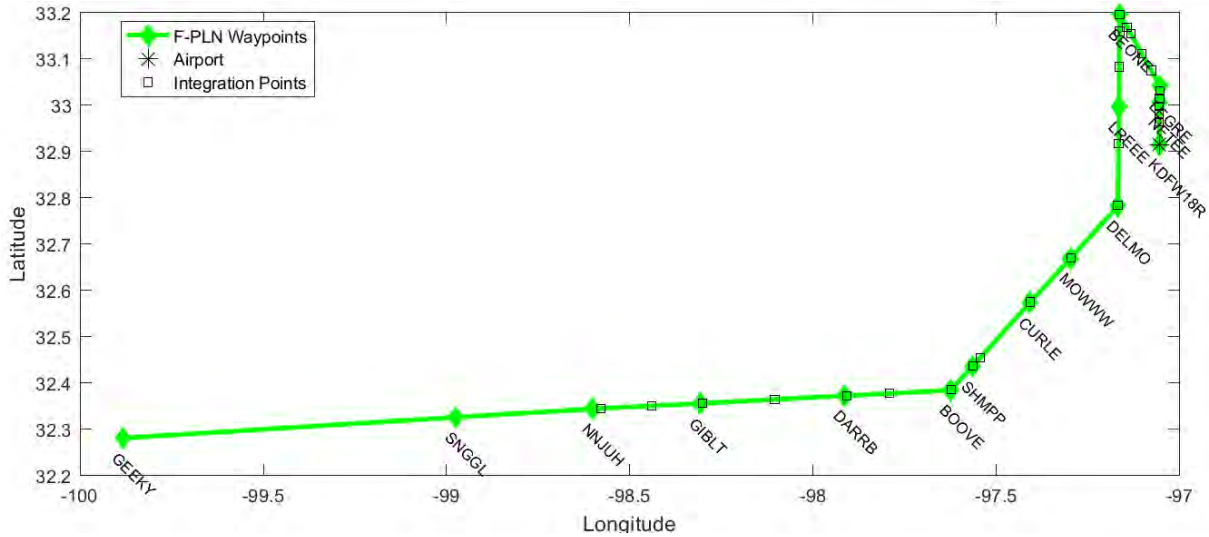


Figure 5.8: Flight plan preparation for KDFW arrival procedure.

The main advantage of the algorithm is its ability to find the global optimum for any arrival procedure. The current case study was computed using the BADA model. The next case study uses a real version of the PDB, which is applied by the FMS to perform the calculations, so that the trajectory can be compared with the one computed by the FMS. The analysis of the comparison highlights the benefits of the proposed trajectory construction instead of traditional Mach/CAS descent profiles. The gas emissions [102], expressed in kilograms, computed by the model described in appendix C are displayed in Fig. 5.9:

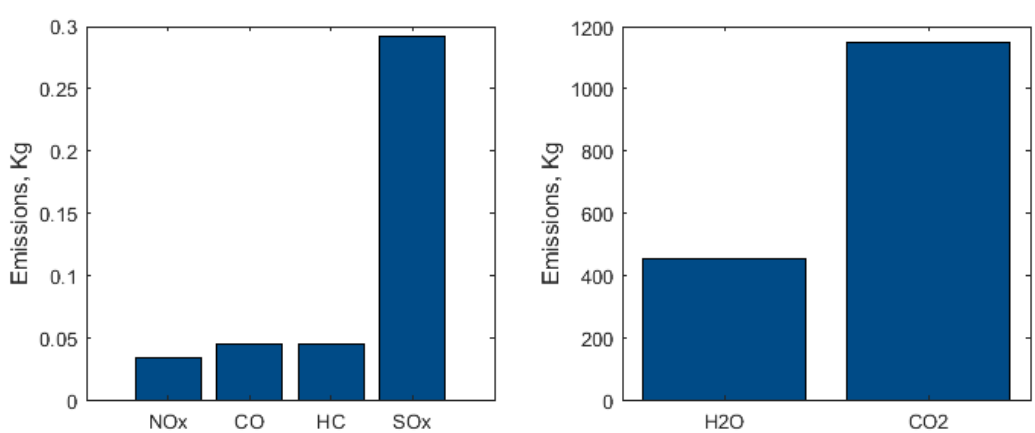


Figure 5.9: Gas emissions comparison for KDFW case.

5.3 Case study II: Los Angeles (KLAX) arrival

5.3.1 Selection of the arrival procedure

The second use case consists in finding an optimal path for the STAR procedure (SEAVU2) corresponding to an ILS approach to runway 24L at Los Angeles airport (KLAX, see Fig. 5.10) with entry point at SEAVU waypoint (refer to Fig. 5.11). The interest of this arrival procedure is linked to the number of altitude constraints and the fact that track variations from the entry point until the runway are minimal. In this case, the resulting trajectory is compared with the profile computed by a certified FMS, and the operational validity of the trajectory has been assessed in the flight simulator. Table 5.4 summarizes the initial and final input states for the calculation, which have been initially produced by the FMS:

	Initial state	Final state
Distance to destination, NM	-2.95	-116.5
Altitude, ft	1125	33000
Speed, kt	133.8	235
Flap settings	Full	Clean

Table 5.4: KLAX case study initial and final conditions.

The list of altitude and speed constraints associated to each waypoint of the procedure is given in Table 5.5:

Waypoint	Type of CSTR	ALT CSTR	SPD CSTR
KONZL	AT	FL170	
ENGLI	AT OR ABOVE	FL160	
PECOX	AT OR ABOVE	FL140	
SEAVU	WINDOW	FL140-FL120	270
PFILA	AT OR ABOVE	FL100	
SALWA	AT OR ABOVE	9000	
WILNUT	AT OR ABOVE	8000	
HURLR	AT OR ABOVE	7000	
FNESE	AT OR ABOVE	6000	
FAYZE	AT OR ABOVE	5000	
JULLI	AT OR ABOVE	4000	
BOUBY	AT OR ABOVE	4000	

Table 5.5: List of constraints with waypoint labels for KLAX case.

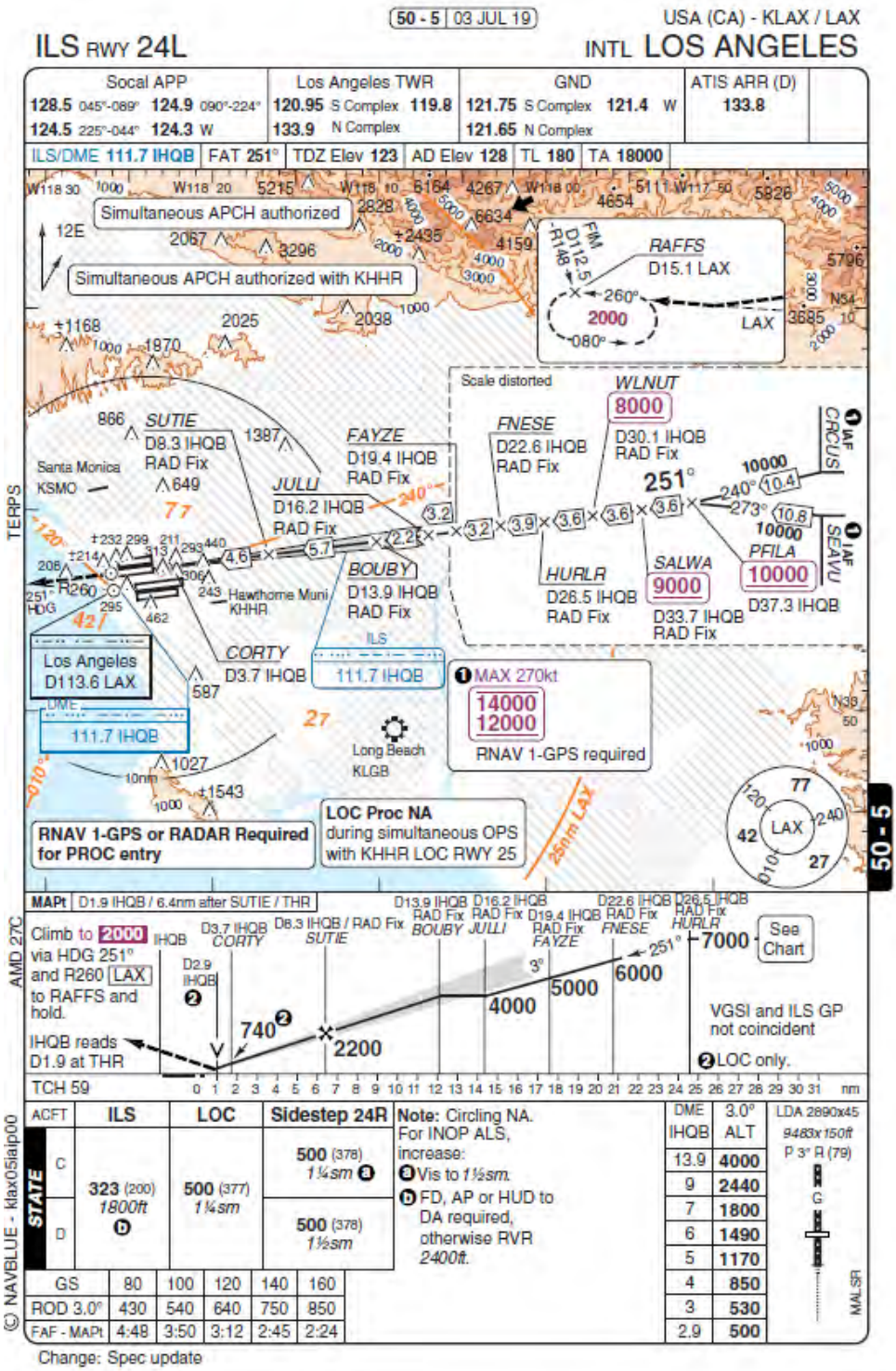


Figure 5.10: APPR ILS-24L. Source: NavBlue.

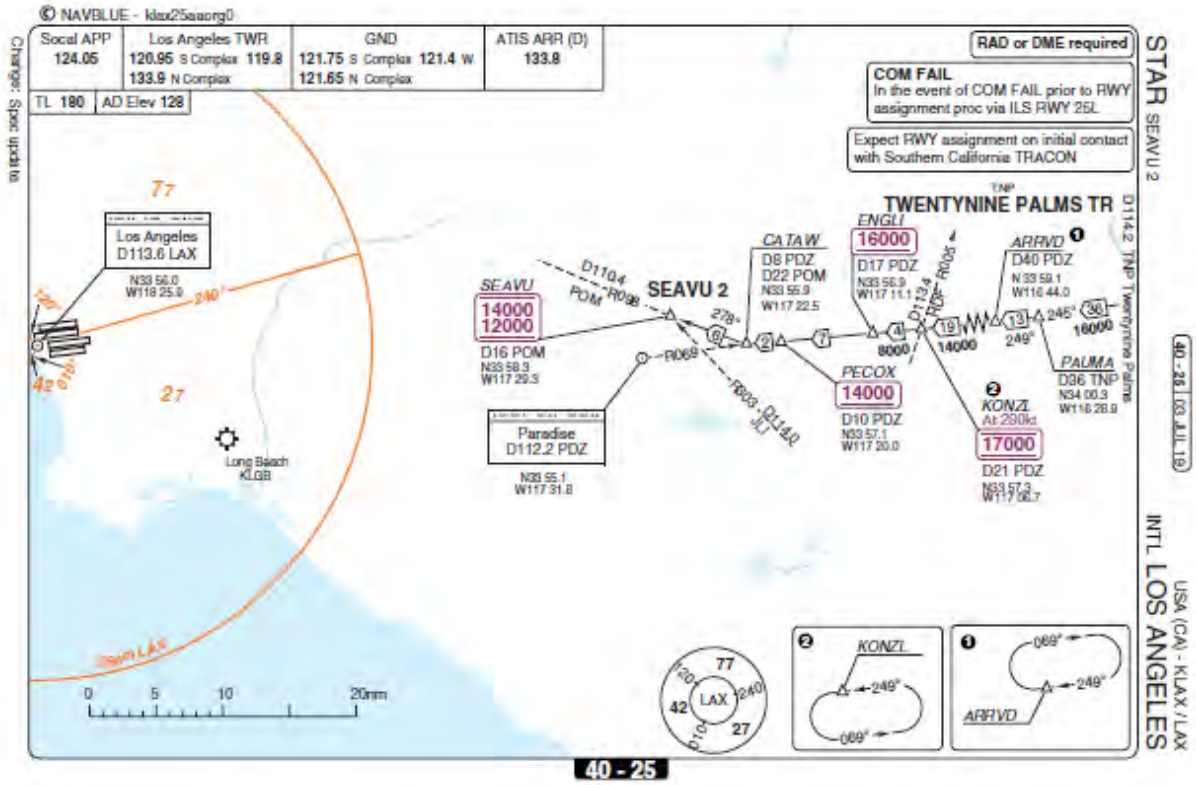


Figure 5.11: SEAVU2 STAR. Source: NavBlue.

5.3.2 Trajectory comparison with a certified FMS

The final state determined by the FMS computation is compared with the calculation provided by the A* algorithm. Nevertheless, A* computes the optimal trajectory for any final state while the FMS always calculates a top of descent, regardless of the actual aircraft position. The altitude and speeds profile computed by the algorithm are compared to the one produced by a real FMS as displayed in Fig. 5.12. A vertical discontinuity is clearly seen at -35 NM in the speed profile (see Fig. 5.13), which occurs when the aircraft is not capable of decelerating and descending simultaneously while satisfy the constraints. For the A* trajectory, initially the aircraft remains in landing configuration until eventually it accelerates to 270 knots in two consecutive level-offs (see Fig. 5.13) at the same time as the altitude constraints are satisfied. On the contrary, the FMS constructs a geometric segment with a shallow path, which fails to accelerate to 250 knots. The geometric segment is defined by an altitude constraint, which defines a fixed flight path. In the case the flight path is shallow, extra thrust is usually required to follow the path and aircraft decelerate on path. If the flight path is steep, the thrust is similar to idle so that aircraft decelerates slowly on the path, since most energy is dedicated to follow the path instead of decelerating.

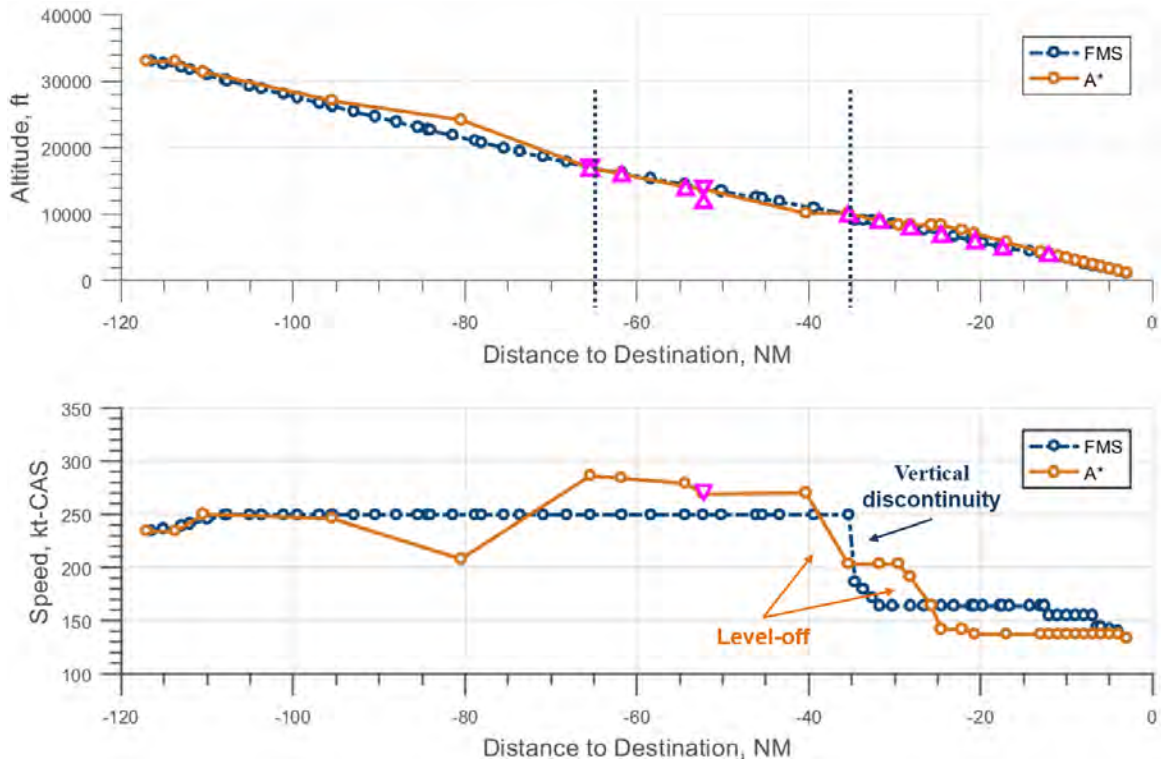


Figure 5.12: Altitude and speed profile comparison between the reference profile computed by the FMS and the optimal one computed by the A* algorithm.

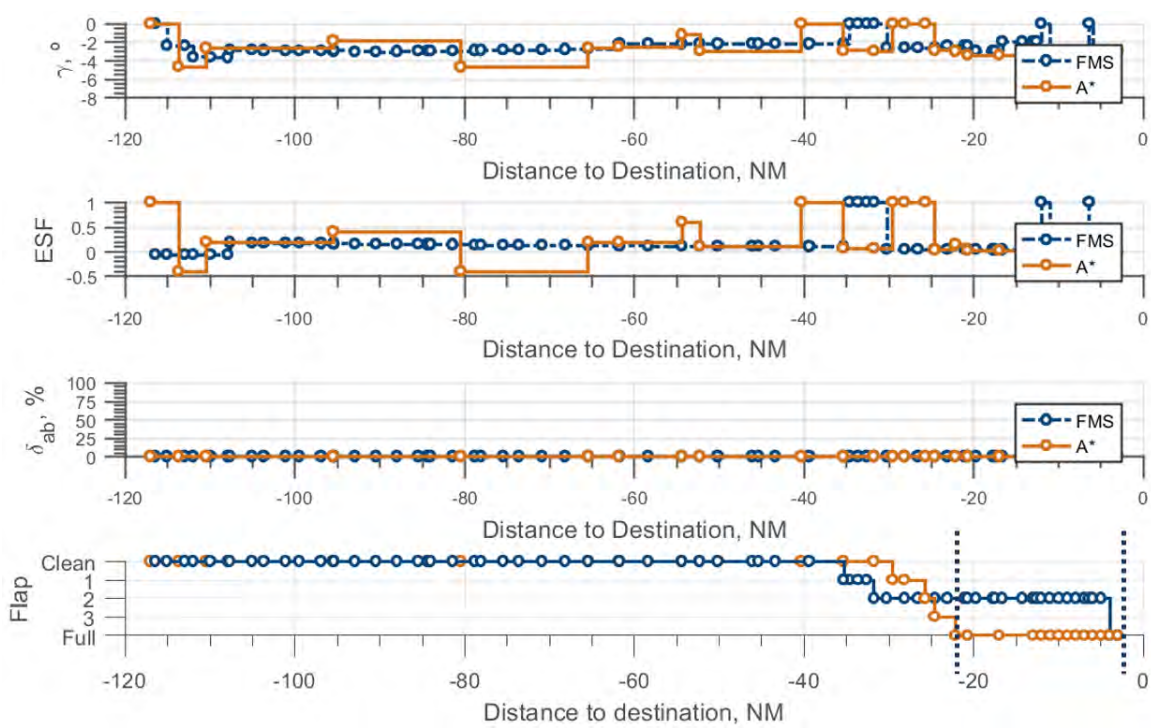


Figure 5.13: Control variables values comparison between the reference profile computed by the FMS and the optimal one computed by the A* algorithm.

The computation produced by the A* algorithm reduces fuel consumption by 13% with respect to the FMS. As a side effect, arrival time is as well decreased by 1% with respect to the FMS reference profile, since the aircraft flies faster close to the speed constraint. In general, fuel savings are well localized; in this case, the shorter approach path and the removal of the geometric path, which occurs between -35 and -65 NM as shown in Fig. fig:KLAXprofile, during the descent are the main causes of this decrease. The A* design splits geometric paths into several segments, which yields a more efficient energy repartition that maintains thrust idle as long as possible, and only adds thrust at the most favorable altitudes.

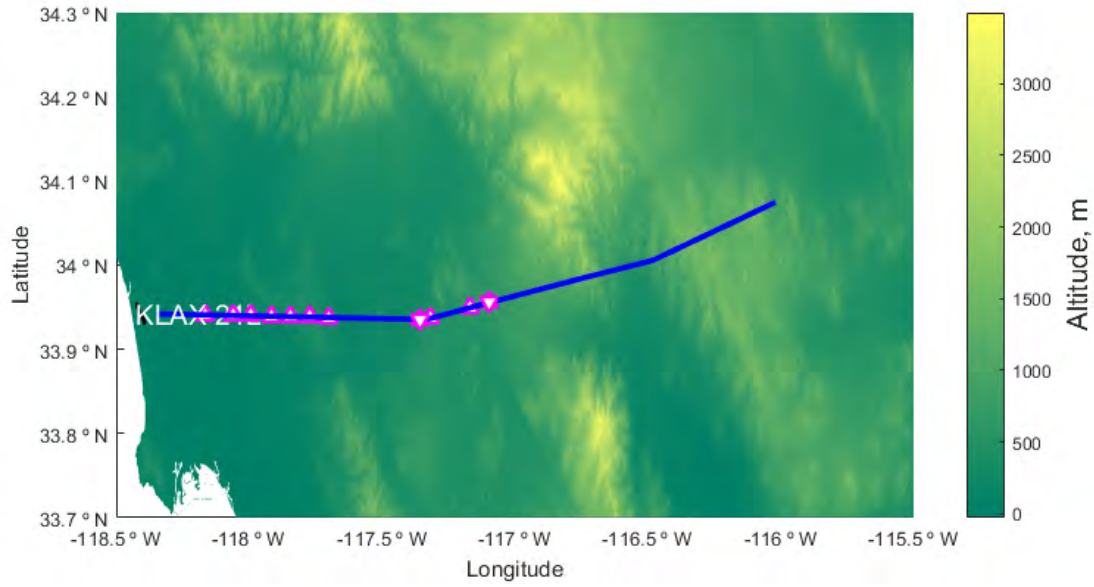


Figure 5.14: KLAX case study: Lateral path visualization.

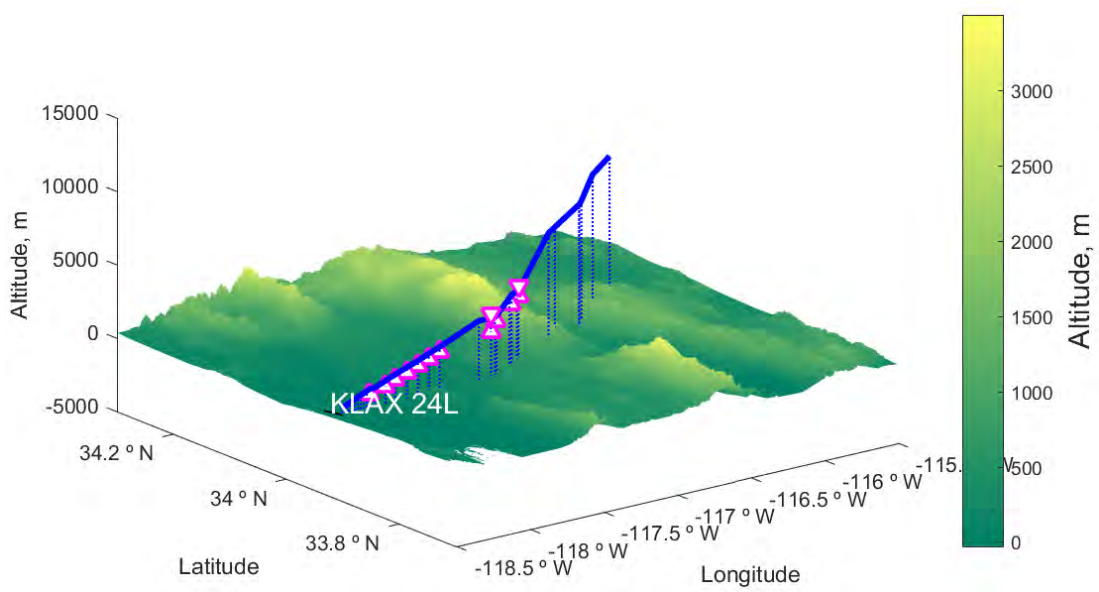


Figure 5.15: KLAX case study: Vertical path visualization.

The trajectory computed by the algorithm can be coupled with the lateral path defined by the procedure, which results in Fig. 5.14. Then, the vertical path trajectory has been displayed in Fig. 5.15, where digital terrain data has been retrieved from [101]. It can be noted that the computed trajectory passes as close as possible to the upper altitude constraint. Finally, the gas emissions of both trajectories has been computed and compared in Fig. 5.16 using the model described in appendix C. It shows that fuel savings also produce gas emissions reduction.

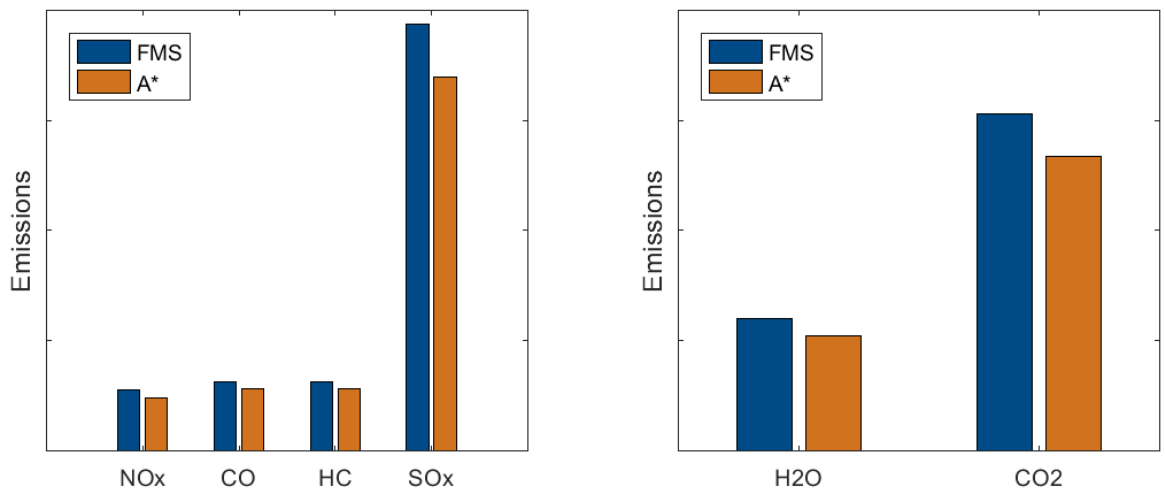


Figure 5.16: FMS and A* gas emissions comparison for KLAX case study. Note that, for confidentiality reasons, the computed values can not be disclosed.

In particular, NO_x has been reduced by 13%, CO and HC by 11% whereas SO_x, H_2O and CO_2 have been reduced by 12.6%. In the current case study, the aircraft flies as high as possible and geometric segments are replaced by several idle paths, which requires less thrust and, hence, less fuel to burn. CO_2, H_2O, NO_x, CO and HC [103] are proportional to fuel consumption, although ambient conditions have an impact as well. Besides, gas emissions are sensitive to the altitude; high altitudes usually reduce NO_x emissions[104], whilst the effect of other gas emissions such as CO and HC is strictly limited to low altitudes [103]. The previous results demonstrate that there is strong correlation between gas emissions and fuel consumption, where the former has been reduced by 12%, on average, and the latter has been reduced by 13%. Nevertheless, the proposed trajectory does not guarantee the minimization of gas emissions, since there could be other trajectories that decreases these figures. For doing so, gas emissions could be considered as the optimization function instead of simply a variable dependent on fuel consumption. Nonetheless, that study is out of the scope and fuel consumption is considered as the main optimization variable in descent and approach since, sometimes, flight time and gas emissions are optimized collaterally.

5.3.3 Assessment of the trajectory in A320 flight simulator

The computed trajectory was flown in a A320 flight simulator at Airbus facilities to check that the behavior of the aircraft was consistent with the calculations. The integration points used by the algorithm were entered manually into the flight plan through their latitude and longitude coordinates, which helped to change the flight path targets at the correct distance for the trajectory monitoring. The arrival procedure is shown in Fig. 5.17, where the STAR procedures waypoints are displayed by green-diamonds and those computed by the A* are displayed by black-squared points. These waypoints were stored in the FM memory and entered into the flight-plan.

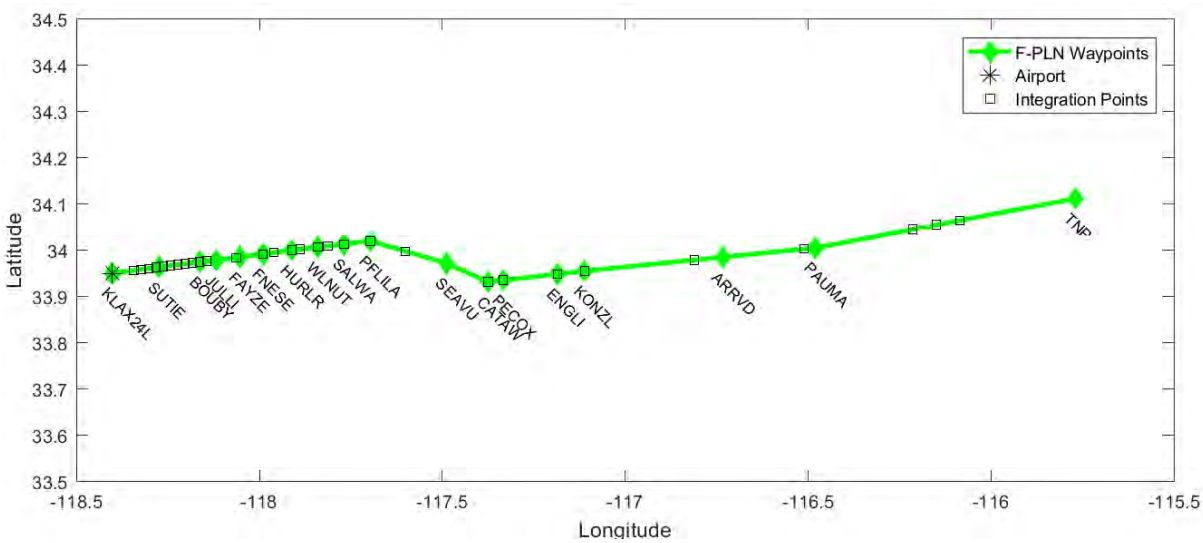


Figure 5.17: Flight plan preparation for KLAX arrival procedure.

As of today, there is no guidance mode that follows automatically the computed trajectory. Thus, the trajectory was flown with auto-thrust off, thrust levers manually adjusted at idle setting and the auto-pilot switched on, in order to follow the lateral path. Aircraft vertical motion was managed by means of successive FPA adjustments. An example of the auto-flight and thrust configuration that was used during the tests is given in Fig. 5.18. Auto-thrust was deactivated since auto-pilot gives priority to speed targets instead of altitude targets, which means that aircraft decelerate or accelerate to the target speed before following the vertical path. The trajectory was followed properly through successive FPA changes at the proper integration points. The “Pilot Flying (PF)” adjusted the FPA values on the FCU whereas the pilot “Pilot Non-Flying (PNF)” checked that the changes were done at the proper distance and that both altitude and speed profiles were followed. It has to be noted that neither the PF nor the PNF are professional or flight test pilots. As a general principle, transitions from shallow to steep paths were anticipated to limit

over-shooting due to flight controls inertia.

(a) Flight Control Unit (FCU) adjustment.

(b) Idle thrust levers.

Figure 5.18: Auto-flight and thrust levers setting for the simulation tests.

The comparison between the trajectory calculated by the algorithm and that flown in the simulator is given in Figs. 5.19 and 5.20 as a function of time. The speed profile observed in Fig. 5.20 is not exactly the same as the one of Fig. 5.12, since it corresponds to a previous calculation where the 270 knot speed constraint was misplaced with the purpose of observing higher speed variations. In Fig. 5.12, speed varies little compared to Fig. 5.20, where speed decreases and increases consecutively up to three times.

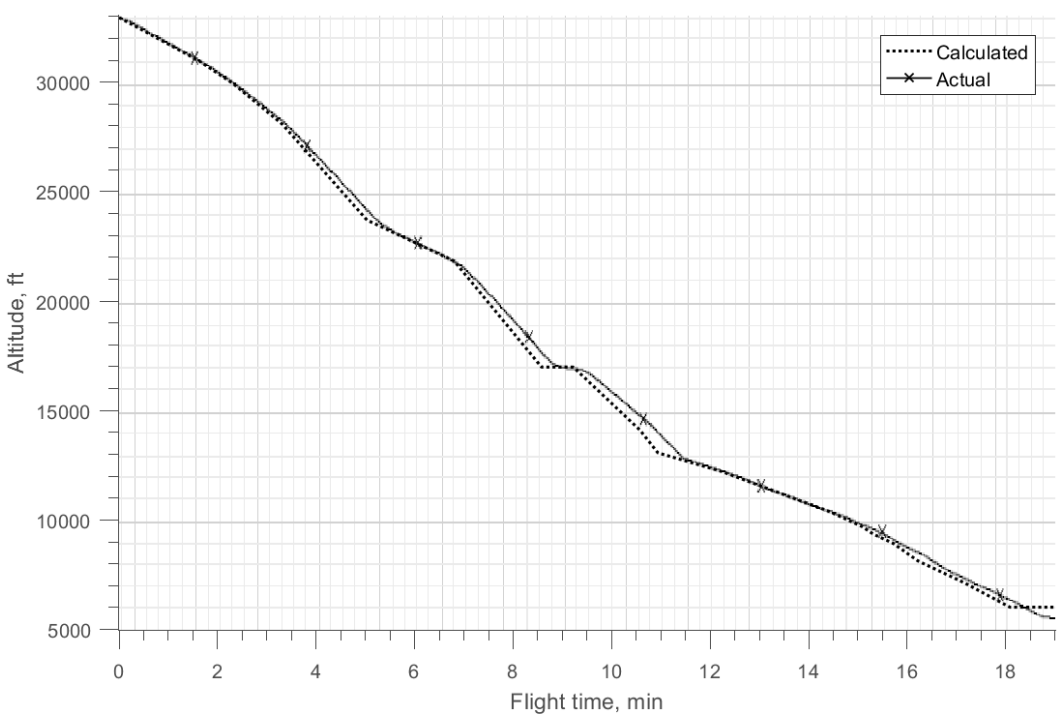


Figure 5.19: Altitude profile comparison between the computed trajectory and the actual one flown in the flight simulator.

Whereas the altitude profile was followed correctly as shown in Fig. 5.19, speed deviations are observed in Fig. 5.20; the aircraft decelerates as calculated but accelerates less than expected. This is likely due to the existence of idle margins that over-estimate the actual idle rating of the engines, which are only used for the calculation of the profile. Thrust was manually added to compensate that lack of acceleration, as it can be observed in Fig. 5.20 between minutes 11 and 12. The altitude error at 5000 ft is about 400 ft, which is relatively small compared with the speed error of 8 knots, approximately. From an operational perspective, this preliminary assessment suggests that the concept is not easy to put in operation today, due to the increased workload resulting from the continuous changes of FPA target, but is physically flyable from a performance perspective. The design and implementation of a guidance mode on the basis of the energy-sharing concept could enable this type of flight operations. From an ATC perspective, the acceptance of variable optimal speed descent profiles instead of traditional Mach/CAS may depend upon the implementation of the trajectory information sharing between on-board and ground.

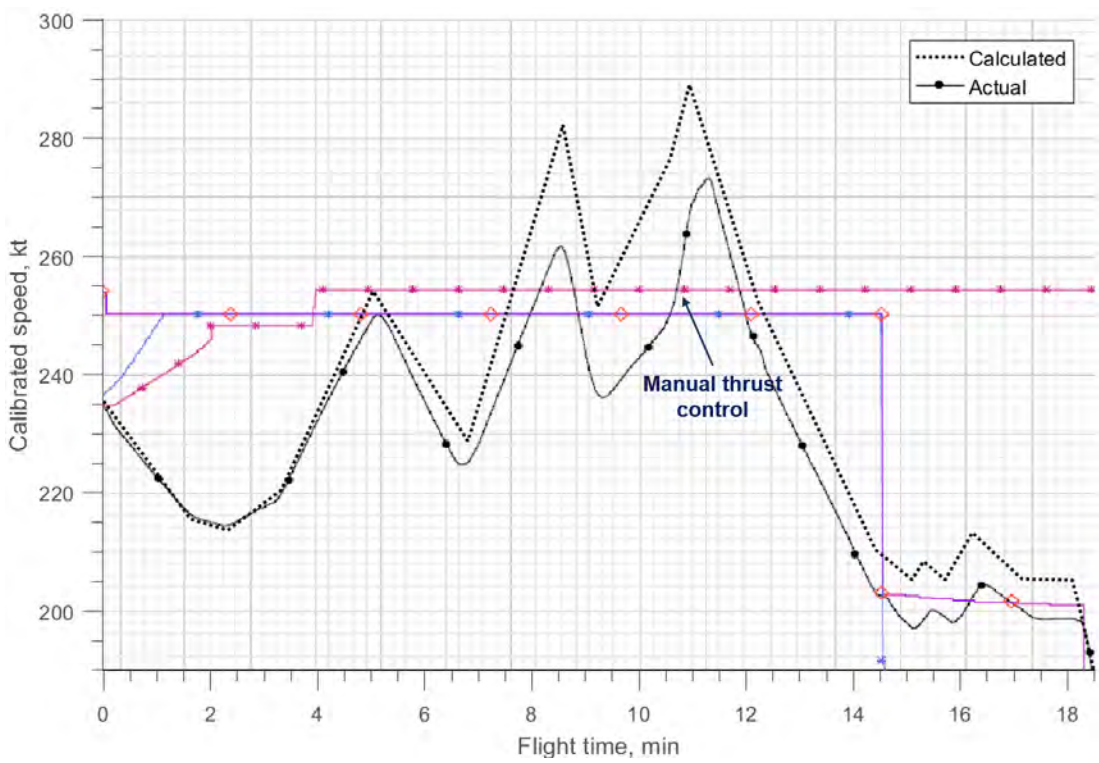


Figure 5.20: Speed profile comparison between the computed trajectory and the actual one flown in the flight simulator simulator.

It is noted that, in Fig. 5.20, the computed trajectory has been superimposed to flight simulator data. These data is provided by the FMS and contains: speed target sent to PFD for display (pink line), calibrated speed target (purple line) and theoretical speed on path (blue line). This information is disregarded as it has no impact on the comparison.

5.4 Case study III: Aircraft high-energy condition

5.4.1 Aircraft high-energy condition in approach

The approach is one of the most complex and labor-intensive phases of flight for flight crews. Flight crews manage the energy of the aircraft through the flight controls. In managed modes, pilots supervise continuously the altitude and the speed of the aircraft and use thrust or airbrakes to compensate any energy error. Besides, during approach phase, they can anticipate flap changes to generate more drag. In selected modes, specially where crews have deviated from their intended flight plans, they have to estimate the current aircraft energy condition in order to continue the approach [105]. Contributor factors such as unexpected weather conditions, ATC radar vectoring techniques or degradation of the aircraft performance may increase the difficulty of the operation [22]. Additionally, the FMS gives little assistance when the aircraft is off-path, since the system does not know when and where the aircraft will return to its intended path. A high-energy condition, in which the aircraft is too fast, too high or both, could trigger a non-stabilized approach [23]. The re-computation of the flight strategy during the approach phase would be beneficial for pilots, as this new trajectory would help them to manage the energy of the aircraft.

This case study assumes that the aircraft is close to the runway threshold and relatively high and fast to be considered in high-energy condition. Under theses circumstances, flight crews shall increase the energy dissipation rate of the aircraft to reduce the energy and ensure stabilization at the 1000 feet gate. The previous calculations were focused on strategic flight planning as the aircraft was not in descend or approach yet. However, in this case the algorithm provides tactical adjustments to solve, on a real-time basis, the current aircraft over-energy. The parameters for the simulation are described in Table 5.6:

	Initial state	Final state
Distance to destination, NM	-3.14	-19.5
Altitude, ft	1500	7500
Speed, kt	129.5	250
Flap settings	full	clean

Table 5.6: Initial and final states used for the computation of the high-energy case study.

In this situation, pilots usually apply a set of rules to estimate the best strategy to dissipate the energy excess and act consequently on the control devices. The success of the operation depends on the flight crew expertise and knowledge of the aircraft performance but also on the weather conditions. An excessive energy dissipation may result in a

low-energy condition and require additional thrust, which decreases the efficiency of the flight. The worst case is the one where safety is compromised as a result of an energy mismanagement and could lead to a Controlled Flight Into Terrain (CFIT).

5.4.2 Flight strategy comparison

In order to assist pilots in energy management and contribute to the safety and efficiency of the flight, the algorithm proposed in this thesis is able to compute the trajectory that solves the high-energy condition. In the case that no solution is found, the aircraft shall stretch the lateral path to ensure a stabilized approach. This topic will be addressed in the next chapter with the computation of the high energy-limit trajectory. Regarding the resolution of the high-energy condition during approach, different trajectories solve the over-energy situation and stabilize the aircraft before landing, whose choice depends on the optimization criterion. In this section, two trajectories as a result of different flight controls combinations are presented.

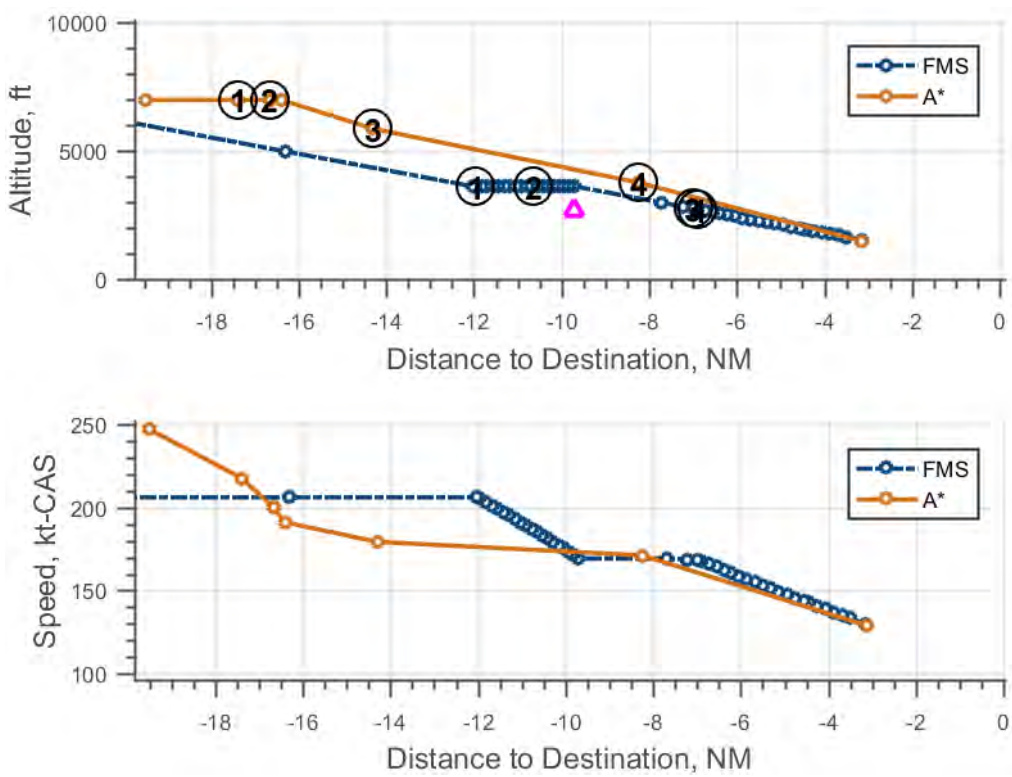


Figure 5.21: Flight strategy 1: Altitude and speed profiles. The aircraft is too fast and too high. Flap configuration extensions are marked with numbers on the altitude profile. Priority given to deceleration.

In general terms, flight strategy 1 anticipates flap changes and limits the use of airbrakes, as required. Flight strategy 2 relies on airbrakes extension as primary source of

drag generation whereas flap configurations are delayed and changed at lower speeds. Note that the FMS trajectory is displayed by the dashed-blue line, which represents the reference trajectory computed by the system without solving the high-energy situation. The objective function to optimize is different for both strategies. The first strategy aims to minimize airbrakes extension along the trajectory whereas the second one minimizes the fuel consumption.

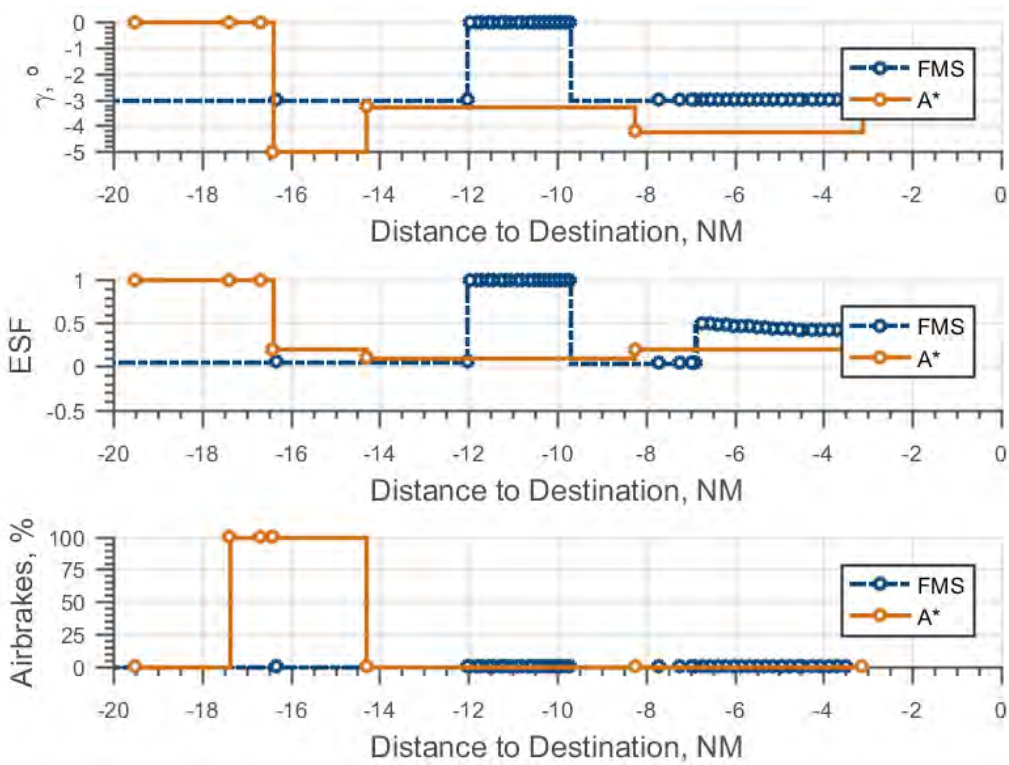


Figure 5.22: Flight strategy 1: Flight control values. The extension of airbrakes is minimized while flap extensions are anticipated.

On one hand, in the strategy 1, priority is given to speed, as it can be seen in Fig. 5.21. The aircraft decelerates in level-off and extends flap configuration as soon as possible. In the transition between the level-off and the descent segment, airbrakes are extended to increase the rate of descent and decelerate at the same time (see Fig. 5.22. Then, flap 3 is set and the aircraft decelerates slowly on path as it descends to the stabilization gate. On the other hand, strategy 2 gives priority to altitude as displayed in Fig. 5.23. The aircraft maintains a high speed as long as possible and increases the rate of descent due to the use of airbrakes (half position). Then, the aircraft decelerates in level-off with full airbrakes deployed (see Fig. 5.24); flap configurations 1 and 2 are extended in the meantime. Airbrakes are inhibited as soon as flap configuration 3 is extended, and then the aircraft decelerates on the -3° until reaching the stabilization gate exactly at the approach speed.

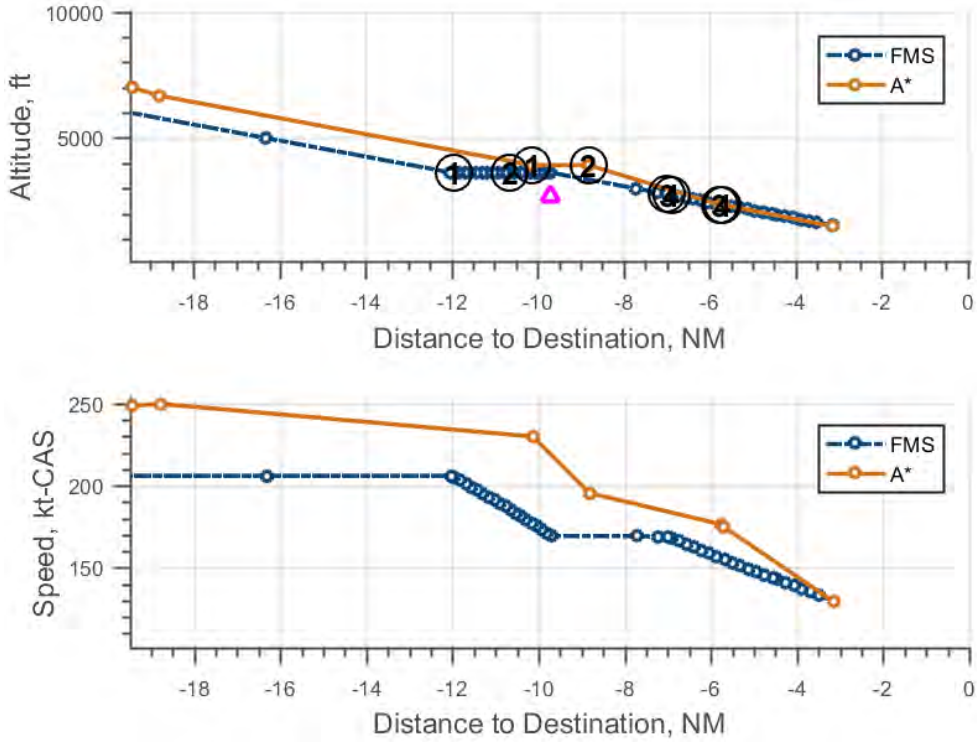


Figure 5.23: Flight strategy 2: Altitude and speed profiles. The aircraft is too fast and too high. Flap configuration extensions are marked with numbers on the altitude profile. Priority given to descent.

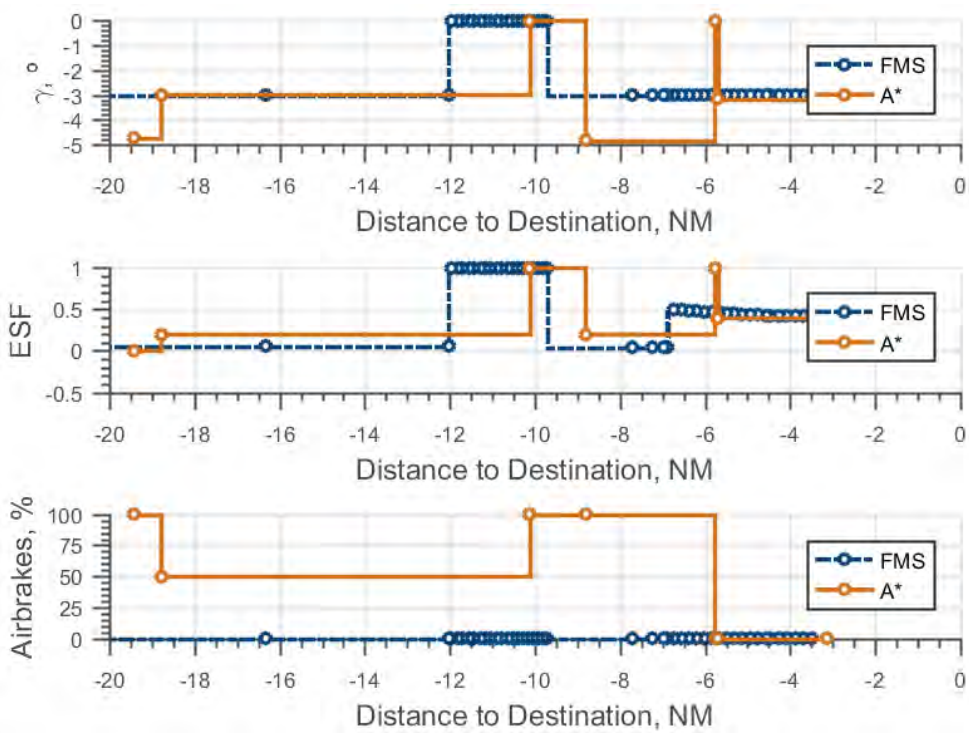


Figure 5.24: Flight strategy 2: Flight controls values. Airbrakes extended while required and flap extensions are delayed.

Results show that flight strategy 1 burns 18% more fuel than strategy 2, since idle ratings increase with flap changes for compliance with engine certification. However, this strategy reduces airbrakes utilization by 55%. Since safety is always the top priority, flight strategy 1 shall be privileged. The is because it provides an additional safety margin, since airbrakes could still be deployed in case of an unexpected tailwind. The anticipation of flap changes combined with a limited use of airbrakes is the preferred option for the design of the function. The extra margin provided by airbrakes may reduce the likelihood of going around, whose impact on workload and flight efficiency is much worse than the fuel difference between both strategies.

In order to improve the design of the function, further assumptions are required. As of today, the real cost of airbrakes extension is unknown; for instance, the relationship between structural degradation and deflection angle is not linear and, hence, difficult to model. Therefore, full airbrakes extension for short periods of time seems to be more penalizing than half extension for longer periods, but this effect is not properly modeled, and it is not consequently integrated in the optimization function. Thus, the final design of this function should account for fuel, airbrakes extension and other operational factors. Among these factors, it could be considered that airbrakes are continuously extended instead of performing several extensions and retractions, even for small deflections, or, similarly, the compromise between time of extension and deflection angle. There is intense debate among experts around this topic and improved models should provide a better knowledge of the real cost behind the airbrakes utilization and, then, converge to an improved design of the function.

5.5 Discussion of the results

The algorithm presented in this thesis computes the optimal trajectory for any arrival procedure according to a certain optimization criterion. For descent and approach phases, the optimization has been focused on fuel consumption, since relevant fuel savings may be obtained and, in some cases, flight time is decreased as well. The obtained results suggest that, in descent and approach phases, fuel minimization should be envisaged, since time can be easily reduced during cruise phase. Besides, gas emissions are proportional to fuel consumption and noise generated from thrust is also reduced, since thrust is set to idle as long as possible. Whenever it is necessary, extra thrust is added at most fuel-efficient altitudes, in general, at higher altitudes than FMS geometric segments.

Regarding the analyzed case studies, on one hand, the computation provides the optimal path to be followed by a commercial aircraft in order to minimize the fuel consumption, which corresponds to the nominal case. On the other hand, there are cases

during the approach where the aircraft is in high-energy condition, which requires other control variables to ensure aircraft stabilization. As soon as this condition is identified, the algorithm aims at minimizing the extension of airbrakes, since it provides an additional safety margin and their utilization has a large impact on maintenance cycles. The first trajectory presented was calculated with a BADA performance model, which could not be compared with the one of a FMS as disparity between models made them not comparable. Therefore, the second and third case studies used a real Airbus A320 PDB for fair comparison. In the second case study, the resulting trajectory suggest that fuel consumption can be decreased by 13% whilst flight time is also reduced by 1% as a side effect. Flight simulation has demonstrated that the computed trajectory can be flown with current guidance modes. However, this operational procedure is not standard and increases workload of flight crews, which requires the development of specific guidance laws to tap the full potential and benefits. In case study 3, the aircraft was in high-energy condition in approach. In these situations, there is strong debate about which variable has to be optimized. Results suggest that the optimization should focus on the utilization of airbrakes, since a lower solicitation would provide an additional safety margin and reduce noise emissions, although fuel consumption is increased by 18% as a result of anticipated flap changes. The study concludes that modeling improvements on the the utilization of airbrakes should facilitate the definition of a more representative objective function that would take into account the combined cost of fuel and airbrakes.

As observed from the case studies introduced in this chapter, the algorithm generates fuel-efficient trajectories through enhanced energy management, which is based on the continuous exchange between kinetic and potential energy, and results in speed-variable descent profiles rather than traditional Mach/CAS profiles. The A* algorithm finds the optimal control values that generate those states which are part of the optimal trajectory. These combination of control values defines the optimal flight strategy for a given arrival procedure. It differs from the construction of the descent and approach path provided by the FMS, where these control values are based on fixed hypotheses and may lead to situations in which the trajectory is far from being optimal. A clear consequence of the improved energy management is that vertical discontinuities in the flight plan are solved, even without the need of airbrakes extension. Thus, the current design proposed by the FMSs is improved, since the trajectory is more adapted to the particular STAR procedure. Moreover, the trajectory, which is computed upstream from the destination runway until the current aircraft position, is permanent in the sense that it always reaches the aircraft position regardless of the phase of flight and guidance modes. This is especially interesting during the approach phase, where the algorithm recomputes the optimal strategy to stabilize the aircraft. This real-time re-computation could contribute not only to improve the efficiency of the flight but also to increase the safety of the flight, since the number of

non-stabilized and non-compliant approaches could be potentially reduced. Besides, the proposed function may pave the way to advanced avionics functionalities in light of more automated operations. However, there are still scenarios where a solution does not exist because there is no flight strategy that dissipates the excess of energy and stabilizes the aircraft in the remaining distance to the runway. In these cases, the algorithm explores all candidate nodes of the search space before identifying the ones where no solution exists. This particular situation in which a solution can not be found is addressed in the next section.

6

Energy-Limit Trajectories

The previous chapter introduced a methodology for calculating fuel-efficient trajectories that reach aircraft position. For all the presented case studies, a solution existed. However, some high-energy conditions may result in the absence of a solution, where the algorithm explores the whole graph. This chapter presents a variation of the algorithm that computes, on a real-time basis, the energy-limit strategy, which is defined as the last trajectory that ensures the performance of a stabilized approach. Therefore, no safe solution exists for any aircraft beyond this limit trajectory.

6.1 The energy-limit trajectory concept

6.1.1 Purpose of the function

Safety levels have improved over the years; 2016 [106] registered the least amount of fatal accidents per million departures while 2017 registered [107] the lowest number of fatalities. However, the increased number of operations seems to be followed by a rise of serious incidents, most of them related to runway excursions issues. The distribution of accidents per flight phase for the last 10 years depicts that 35% of fatal accidents[107] occur during approach and landing phases, probably caused by the inherent high workload of these flight phases. Flight crews shall decide to abort the landing and go around as soon as the aircraft is not stabilized for landing [108]. On average, go-around procedures occur around 1 to 3 times every 1000 flights according to the analysis presented in [109], which was conducted by the Flight Safety Foundation. Although the occurrence rate indicates that this type of operation is rare, the study suggests that compliance with go-around policy could have prevented some accidents in the past. In general, go-arounds are safe procedures that increase the workload of both pilots and ATC, reduce the capacity of the airport terminal area and contribute to the propagation of delays. The energy-limit trajectory could bring benefits to flight crews and ATC, since it improves energy awareness and trajectory-predictability. Thus, it is expected that this concept reduces the number of non-stabilized approaches and anticipates go-around maneuvers, so it minimizes operational and economic impacts of this type of operation.

While the case studies presented in the previous chapter handled situations in which the algorithm found the optimal trajectory, in other situations, there is no flight strategy that dissipates the excess of energy and stabilizes the aircraft before landing. This situation is displayed in Fig. 6.1, where numbers 1,2,3,4 represent the computation of flap extensions that correspond to configurations 1,2,3 and full respectively, whose trajectory is independent of the current aircraft location. These situations are likely to occur as a result of unexpected strong tailwinds, poor radar vectoring techniques or inefficient energy management [21][22][23]. A provision of energy information to the flight crew could improve the energy awareness on a real-time basis, with the aim of anticipating the actions and reducing the number of missed approaches. On that basis, the purpose of the function presented in this chapter is twofold:

- On one hand, it provides pilots with a visualization of the limit distance from where a stabilized approach is no longer possible.
- On the other hand, it provides pilots with an advisory of the minimum path distance

to stretch that enables crews to stabilize the aircraft before landing, in cases where the aircraft is already within the non-stabilization boundaries.

The algorithm presented in this chapter is an extension of the A* algorithm presented in previous chapters. It computes the energy-limit trajectory, which is defined as the trajectory that stabilizes the aircraft in the minimum ground distance (i.e. accounting for winds), and then provides the maximum energy dissipation. In other words, given the current aircraft altitude and speed, it is the last trajectory that stabilizes the aircraft, since it provides the maximum energy dissipation, and no stabilization is possible for any distance below the limit threshold.

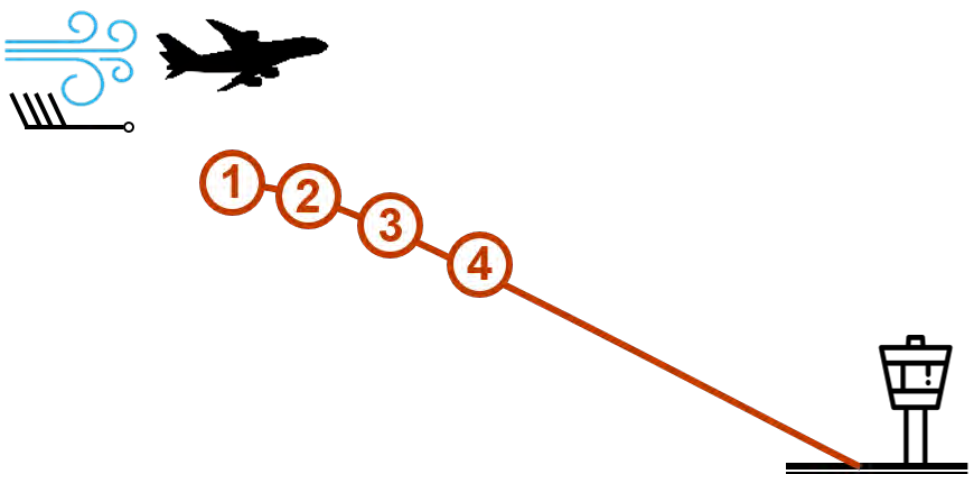


Figure 6.1: Non-stabilized approach as a result of strong tailwinds.

6.1.2 Mathematical formulation

In previous chapters, the optimization criterion was the minimization of fuel consumption. Here, the optimization criterion is the ground distance, since it provides the minimum distance for which the aircraft can be stabilized. The output of the algorithm is a (minimum) distance. Therefore, the independent variable is no longer the ground distance s but the altitude h, whose value is bounded between the initial altitude at the stabilization gate (h_0) and the current aircraft altitude (h_f). The objective function is defined by the following equation:

$$J = \min \int_{h_0}^{h_f} \frac{V \cos\gamma + V_w}{V \sin\gamma} \, dh \tag{6.1}$$

It can be noted from Eq.(6.1) that, in case of zero winds ($V_w = 0$), the objective

function results in the following expression:

$$J = \min \int_{h_0}^{h_f} (\tan\gamma)^{-1} \, dh. \tag{6.2}$$

Since the independent variable is the aircraft altitude, the distance-dependent equations of motion shall be re-written as a function of the altitude, which results in the following formulation of the equations of motion:

$$\begin{cases} s' = \dfrac{ds}{dh} = \dfrac{V\cos\gamma + V_w}{V\sin\gamma} \\ V' = \dfrac{dV}{dh} = \dfrac{g_0 \; ESF \; \gamma_T}{V\sin\gamma} \\ m' = \dfrac{dm}{dh} = \dfrac{-FF}{V\sin\gamma} \\ t' = \dfrac{dt}{dh} = \dfrac{1}{V\sin\gamma} \end{cases} \tag{6.3}$$

The previous equations of motion described in Eq.(6.3) define the state vector of the problem:

$$x = \{s, V, m, t\} \tag{6.4}$$

Then, the control vector contains the control values that generate those states:

$$u = \{ESF, \delta_{ab}, \text{Confaero}, \Delta_{LG}\} \tag{6.5}$$

In previous Eq.(6.5), the variable γ_T does not appear explicitly because, in high energy conditions, thrust is set idle so that the computation of γ_T is a function of the other control variables. Regarding the equations of motion described in Eq.(6.3), the controls δ_{ab}, Confaero and Δ_{LG} modify the drag force, which is used for the computation of γ_T. In clean flap configuration, aircraft speed is limited between the stall speed (V_{LS}) and the maximum operating speed (V_{MO}):

$$V_{LS} \leq V_{CAS} \leq V_{MO}, \qquad \text{if Conf} = clean \tag{6.6}$$

However, during flap changes, the stall speed decreases and the speed shall remain between V_{LS} and the maximum flap extended speed (V_{FE}):

$$V_{LS} \leq V_{CAS} \leq V_{FE}, \qquad \text{if Conf} \neq clean \tag{6.7}$$

In-between, the $V_{CC_{ap}}$ is the characteristic speed for flap changes. In general, flap changes should occur as late as possible as they increase fuel consumption and noise. Therefore, $V_{CC_{ap}}$ is computed from V_{LS} plus a speed margin that accounts for the mechanical delay of the actuators. However, in the energy-limit trajectory, flap changes may be anticipated to increase the energy rate at an early stage. The ESF distributes the aircraft energy loss ($Thr < D$) between kinetic and potential energy. Aircraft decelerations are proven to be more effective during a level-off so, in order to reduce the number of combinations, only relevant ESF values are kept for the calculation:

$$ESF \in \{ESF|_{V_{CAS=const.}}, ESF|_{FPA=-3^\circ}, ESF|_{\text{level-off}}\} \tag{6.8}$$

Where:

- $ESF|_{V_{CAS=const.}}$ is defined as the energy repartition that enables to maintain current V_{CAS} speed.
- $ESF|_{FPA=-3^\circ}$ is the one that decelerates maintaining a -3° FPA
- $ESF|_{\text{level-off}}$ corresponds to a decelerated level-off.

Regarding the airbrakes, limit-energy strategies require the maximum dissipation of energy, so that full airbrakes generate the highest drag:

$$\delta_{ab} \in \{0, \delta_{ab_{Full}}\} \tag{6.9}$$

Since airbrakes extension increases V_{LS}, their utilization is inhibited for landing configurations 3 and Full:

$$\delta_{ab} = 0, \quad \forall Conf \geq 3 \tag{6.10}$$

For each speed considered by the algorithm, a decision of changing or maintaining the current flap configuration is made:

$$\text{Confaero} = \begin{cases} 1 \rightarrow \text{change flap setting} \\ 0 \rightarrow \text{maintain flap setting} \end{cases} \tag{6.11}$$

Similarly, the landing gear is defined as a boolean whose value depends on its position:

$$\Delta_{LG} = \begin{cases} 1 \rightarrow \text{extended} \\ 0 \rightarrow \text{retracted} \end{cases} \tag{6.12}$$

The landing gear extension is limited to a certain speed:

$$\Delta_{LG} = 0, \quad \text{if} V_{CAS} > 280\ kt \tag{6.13}$$

In order to analyze the aircraft energy dissipation capacity, the specific energy rate ($\dot{E}_{T_s}$) is defined as the derivative of total energy:

$$\dot{E}_{T_s} = \dot{E}_{p_s} + \dot{E}_{k_s} = \dot{h} + \frac{V\dot{V}}{g_0} = \frac{(Thr - D)V}{mg_0} \tag{6.14}$$

It can be deducted from Eq. (6.14) that the higher V is, the lower $\dot{E}_T$ (given that $D > Thr$), so a steeper descent is achieved for the same speed target. Besides, decelerations are more effective during level-off segments as potential energy remains constant whereas the energy budget is dedicated to reduce the kinetic energy. As it can be observed, this formulation is a modification of the mathematical model presented in Chapter 3, since additional assumptions shall be considered for the calculation of the energy-limit trajectory. The next section presents the calculation of the energy-limit trajectory and the conclusions obtained from the assessment in the Airbus flight simulator. In reality, the calculation of the energy-limit trajectory is only relevant in approach phase or during the last phase of the descent (e.g. at 30 NM from the destination runway). The algorithm terminates at the aircraft altitude and speed where the optimization variable is the traveled ground distance according to Eq. (6.1).

6.2 Case study at KLAX airport

6.2.1 Initial parameters of the simulation

This chapters presents a case study corresponding to an approach procedure to runway 24L at KLAX airport. The following table 6.1 gathers the current state of the aircraft to be reached:

	Initial state	Final state
Distance to destination, NM	-2.95	s_f
Altitude, ft	1125	7500
Speed, kt	134.6	250
Flap settings	Full	Clean
Landing gear	Down	Up

Table 6.1: Parameters of the simulation.

In a first time, the influence of the wind in the trajectory has been analyzed as displayed in Fig. 6.2.

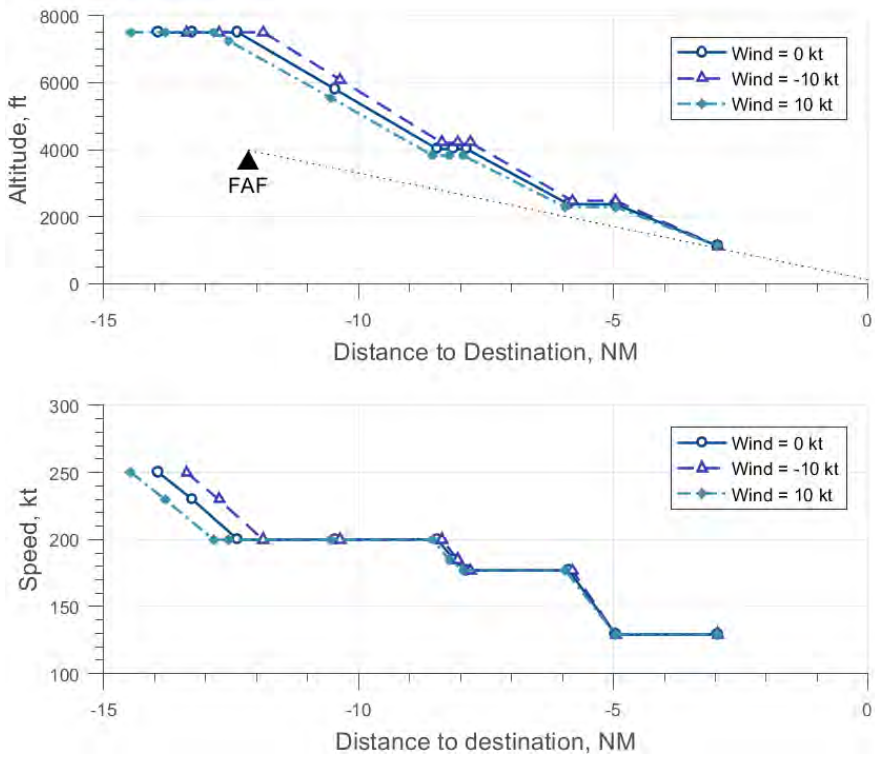


Figure 6.2: Influence of wind in the calculation of the trajectory.

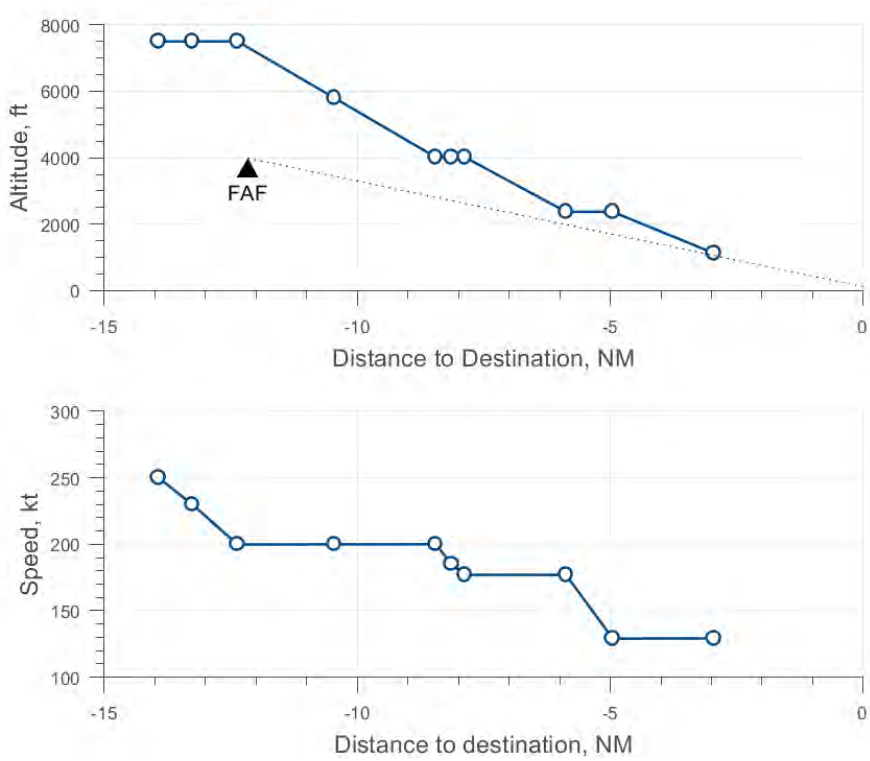


Figure 6.3: Altitude and speed profile for the energy-limit trajectory.

In this case study, the Airbus FMS performance model was used for computing the high energy-limit trajectory. Three constant winds have been tested: zero, 10 knots of tailwind and 10 knots of headwind. This wind value has been conscientiously selected to be representative of the tailwind component limit for take-off and landing defined by the manufacturer for this type of aircraft. Any stronger wind would not be operationally representative for landing. Of course stronger tailwinds could be found at higher altitudes but, since the wind used for the test is constant, this is the maximum selectable value for landing. For this case, it can be observed that a wind of 10 knots increases or decreases ground distance by 4%, depending on the direction of wind.

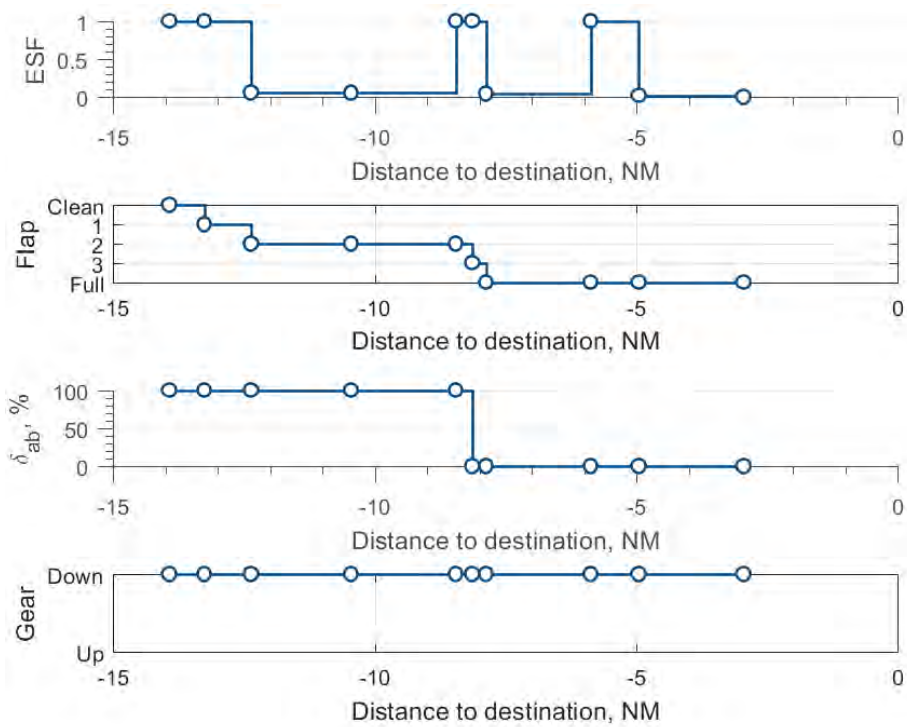


Figure 6.4: Control variables for the energy-limit trajectory.

The analysis of the trajectory is performed for the zero wind case to disregard the wind effect on the minimum ground distance. Figure 6.3 depicts the altitude and speed profiles of the aircraft. The -3 degree glide slope is displayed in dotted line and reaches the altitude imposed by the FAF. Besides, it is observed from Fig.6.4 that the gear is set down from the very beginning of the trajectory and airbrakes are fully extended during clean and flap configurations 1 and 2. The combination of gear down and airbrakes generates more drag to either descend steeper or increase the braking efficiency. Decelerations are performed in level-flight to increase the braking efficiency, and flap settings are changed at maximum speed (V_{FE}) to increase drag as soon as possible. Then, flap configurations 3 and 4 are delayed as much as possible, since the aircraft generates more drag at flap setting 2 with full airbrakes extended than flying in landing setting with airbrakes inhibited. The flight

controls sequence calculated in Fig. 6.4 is visually displayed in Fig. 6.5, in which the speed scale on the PFD is jointly displayed with the flight control actions. The V_{FE} speed is displayed by two parallel amber lines located at 230, 200, 185 and 177 knots, respectively. It is observed that the value of V_{MO} corresponds to that of V_{FE} from the previous step.

Each of the states defined in Fig. 6.5 corresponds to changes of flap configuration, i.e. the actions that occur at distances -13.7 NM, -12.8 NM, -8.2 NM and -7.9 NM (see flap graph in Fig. 6.4). In stages A and B, the gear is down and airbrakes are fully extended whereas in C and D, airbrakes are retracted. It is noted that, the landing gear and flap extensions are irreversible. Since go around procedures are not considered in the calculation, landing gear and flap configurations can not be retracted if they have already been extended.

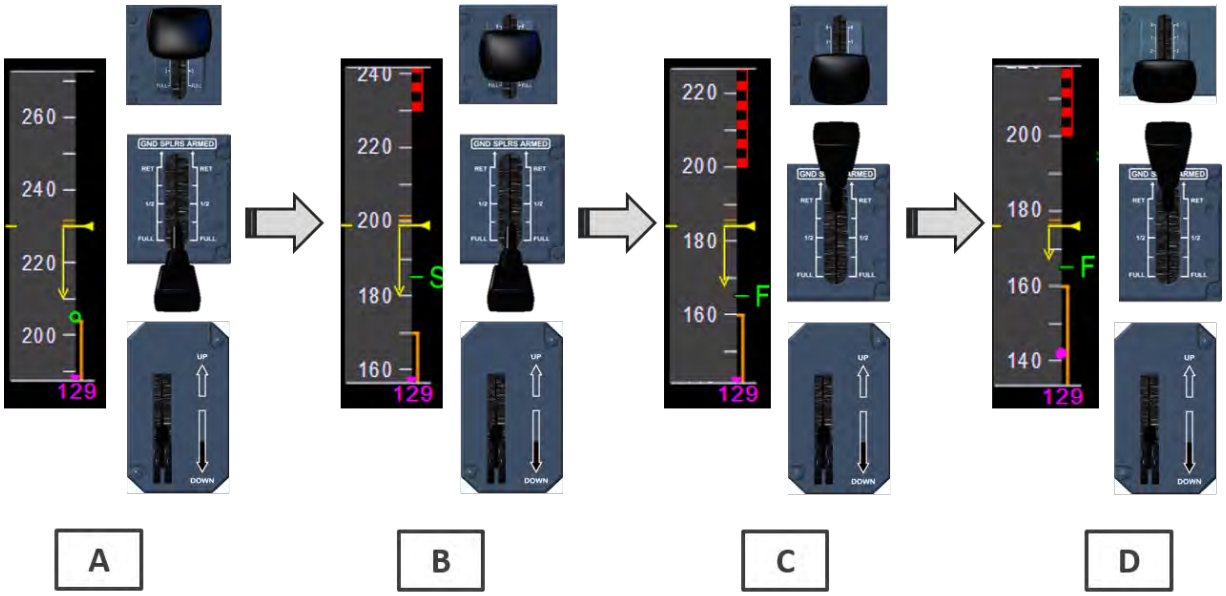


Figure 6.5: Energy-limit flight controls sequence. A, B, C and D represent the sequence of actions to change from clean to landing configuration.

6.2.2 Operational Assessment in the flight simulator

Simulations were conducted at Airbus facilities in order to evaluate the flight operation of the computed trajectory. Several trials were made to acquire precision with respect to the exact positions at which flight control actions should be done. Each scenario needs re-initialization of the flight parameters. These tests were run in the integration bench of an A320 aircraft and took two hours in total. The integration points used by the algorithm were inserted manually in the flight plan under the name of INTAXX (e.g. INTA10, see Fig. 6.6(b)), so the actions were performed at the right moment. Figure 6.6 shows the

EFIS and the MCDU display during the simulator session, the former consisting of a PFD and ND. The auto-thrust was switched-off with thrust levers manually set at idle detent and the autopilot was connected; the aircraft was on path aligned with the runway axis (Fig.6.6(b)) and vertically guided by the FPA targets set in the FCU and displayed on the Flight Mode Annunciator (FMA) of the PFD, as shown in Fig.6.6(a).

(a) PFD display. (b) ND display. (c) MCDU F-PLN page.

Figure 6.6: EFIS and MCDU display during the simulation tests.

The feedback collected after the simulation session shows that the workload was reasonable and the aircraft behaved as calculated, except for slight delays in the FPA changes. The guidance law transition between a shallow FPA (e.g. 0°) and a steep one (e.g. -7°) is not immediate and the aircraft over-shoots the calculated profile. Basically, small over-shoots occurred when transitioning from level-offs to steep paths. Similarly, under-shoots happened when the aircraft changed from -7° FPA to level-off. This confirms the bang-bang behavior of γ that could be avoided by the introduction of γ as a control variable, instead of an intermediate value dependent of control variables γ_T and ESF. A more reactive guidance law could also solve the issue, nevertheless, it would generate undesirable effects on passenger comfort due to excessive normal acceleration.

The operational assessment of the function concluded that intermediate level-offs are undesirable and difficult to be followed by present guidance laws. Thus, the previous calculation of the energy-limit trajectory is somehow theoretical, which needs refinements to improve the operational representativeness and facilitate the path-tracking. Therefore, a more conservative energy-limit trajectory, which accounts for additional operational constraints, has been constructed and tested in the simulators. The proposed trajectory is displayed in Fig. 6.7. This energy-limit trajectory requires 1.5% extra ground distance than the one of Fig. 6.3, which could be considered as a pseudo-optimal solution providing a good compromise between operational representativeness and optimality. The intermediate level-offs have been removed, and only the initial deceleration segment is accepted for initial deceleration. Moreover, it is assumed that the aircraft shall be on the glide path as soon as possible, so that flap changes from 2 to full occur during final approach at -3°

FPA. This design choice aims at minimizing the under and over-shoots produced by sharp changes of FPA, specially when they occur close to the stabilization gate.

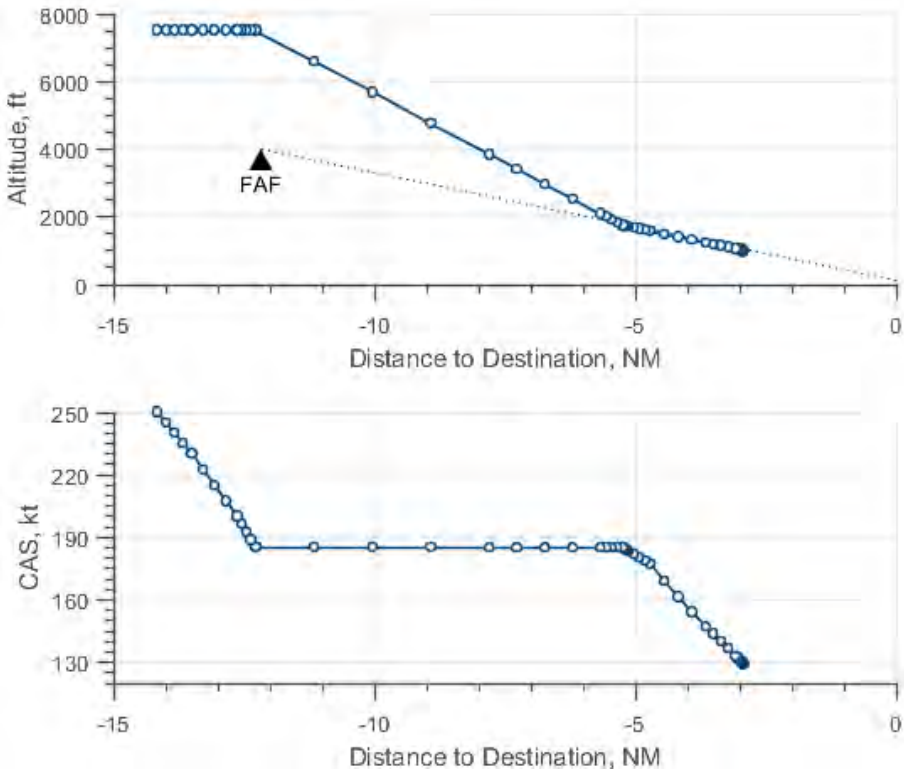


Figure 6.7: Energy-limit strategy sequence.

The use of the A* algorithm for computing the energy-limit trajectory is the appropriate choice for real atmospheric conditions which account for the wind profile, since winds plays a relevant role in the computation of the minimum distance. However, for the particular case of ideal atmospheric conditions (zero wind and standard atmosphere condition), the energy-limit trajectory can be generated simply using traditional integrations with sufficient flight performance knowledge. In order to check this fact, the trajectory displayed in Fig. 6.7 has been generated using Runge-Kutta integrators instead of the A* optimization algorithm. Then, the trajectory has also been tested in the flight simulator. In this case, the trajectory-tracking was easy. The aircraft decelerates in level flight until flap configuration 2 is achieved. Then, the FPA target changes from 0° to -7.7°. An over-shoot occurred as in the previous case, although this time the FPA was slightly increased to -8° in order to compensate the altitude error. Then, 5 NM before the runway, the FPA was changed to -3°, the aircraft decelerated on path deploying flap configurations 3 and Full at the right speeds. The aircraft arrived to the stabilization gate at the computed V_{APP}, although 200 ft above the 1000 ft. In general, the profile was tracked easily, the workload was reduced and the error at the stabilization gate decreased with respect to the previous test. Therefore, although the latter energy-limit trajectory is 1.5% longer than the former

one, it is operationally more representative. Besides, further discussions with flight test pilots may mature more the concept, for instance, over-shooting can be diminished by introducing a transition segment between level-offs and steep descent segments.

6.2.3 Visualization of the energy-limit trajectory

The energy-limit trajectory provides the trajectory that stabilizes the aircraft in the minimum ground distance. This trajectory is a back-up calculation continuously displayed to pilots in order to improve their awareness of the aircraft energy condition. The minimum ground distance provides a useful awareness of the aircraft energy with regard to the runway. On one hand, this distance alerts flight crews of the proximity of the zone from where the stabilization is unfeasible (see left picture in Fig. 6.8). On the other hand, in high-energy cases where stabilization is already unlikely, this distance is used as the minimum to stretch the path for stabilization (see right picture in Fig. 6.8). Flight crews do not need to resume their efforts on stabilizing the aircraft and could directly demand radar vectors to the ATC in order to be reintegrated in the aircraft arriving flow.

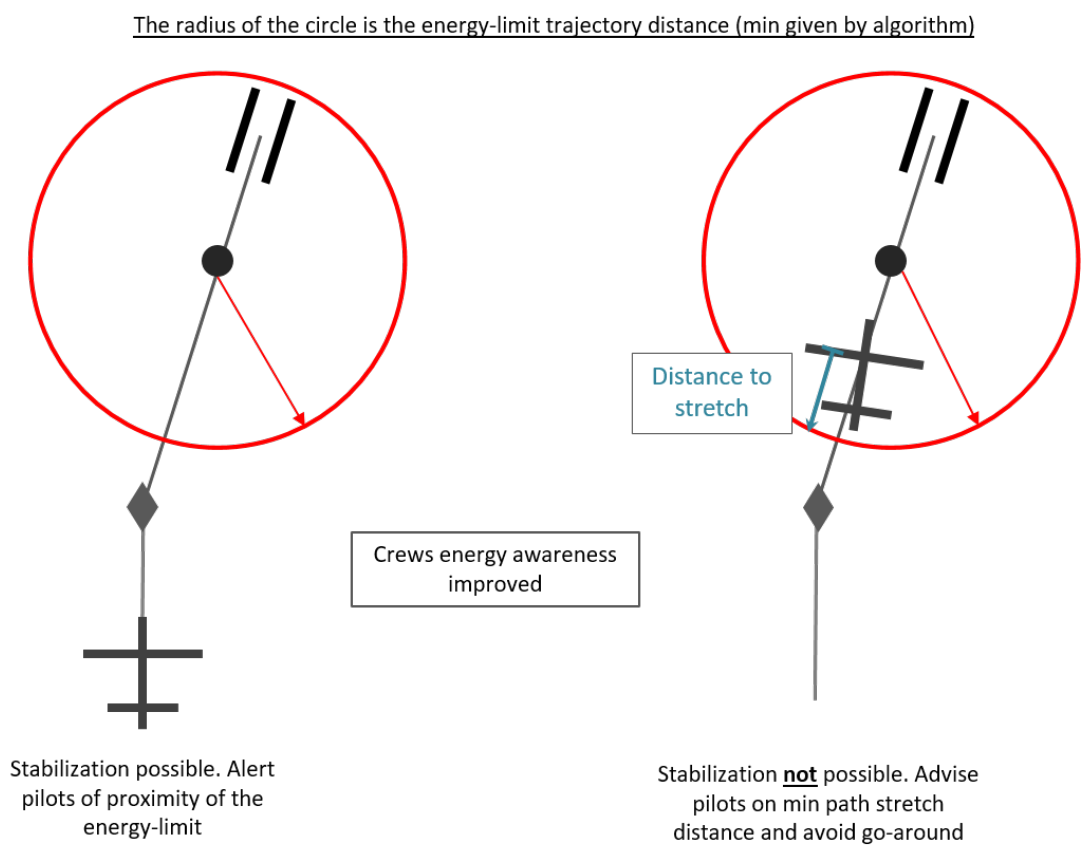
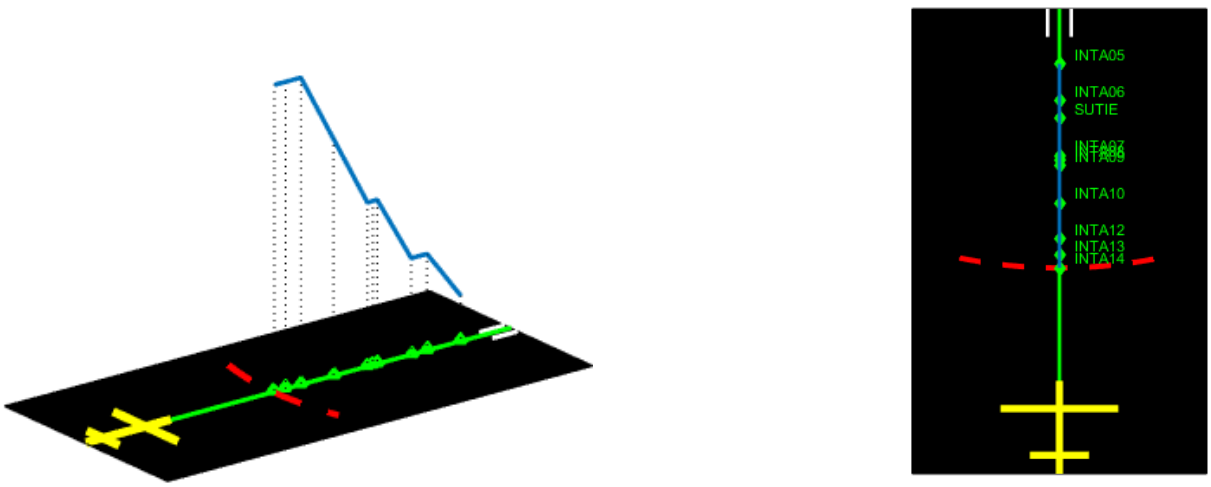


Figure 6.8: Operational use of the energy-limit trajectory. Here, the minimum distance computed by the energy-limit trajectory is displayed as the radius of the circle centered at the stabilization gate.

The previous figure provides two operational cases in which the energy-limit trajectory is valuable. Since approach is a labor-intensive phase for flight crews, especially during complex high-energy situations, the display of the energy-limit trajectory and the minimum ground distance needs to be simple and easy to interpret.

A proposal for the visualization of the energy-limit trajectory is given in Fig. 6.9. The display is based on the ND and the minimum ground distance is shown by the energy-limit arc concept, whose center is at the stabilization gate. The integration points of the vertical trajectory, which are displayed by the blue solid line in Fig. 6.9(a), are projected on the lateral path. The output distance from the calculation, which is the minimum distance to stabilize the aircraft for the given energy condition, is visualized as an arc, whose center is the stabilization gate. On one hand, if the aircraft is behind the energy-arc, stabilization is still possible under the current energy condition. Flight crews margin is reduced as soon as the aircraft gets closer to the energy-arc. On the other hand, the aircraft shall go around as soon as it enters in the energy-arc zone (Fig 6.9(b)), since the stabilization is not possible. In these situations, the position of the aircraft with respect to the energy-circle gives the path distance to stretch in order to ensure stabilization.



(a) 3-D interpretation.

(b) ND alike display.

Figure 6.9: Visualization of the energy-limit trajectory through the energy-limit arc.

The lateral and 3D representation of the energy-limit trajectory by virtue of the energy-circle concept for KLAX airport is provided in Figs. 6.10 and 6.11. The FAF waypoint and the -3 degree glide path of final approach are also displayed, which highlights the resulting steep path of the energy-limit trajectory. The terrain data has been retrieved from [101]. The arc is centered at the stabilization gate.

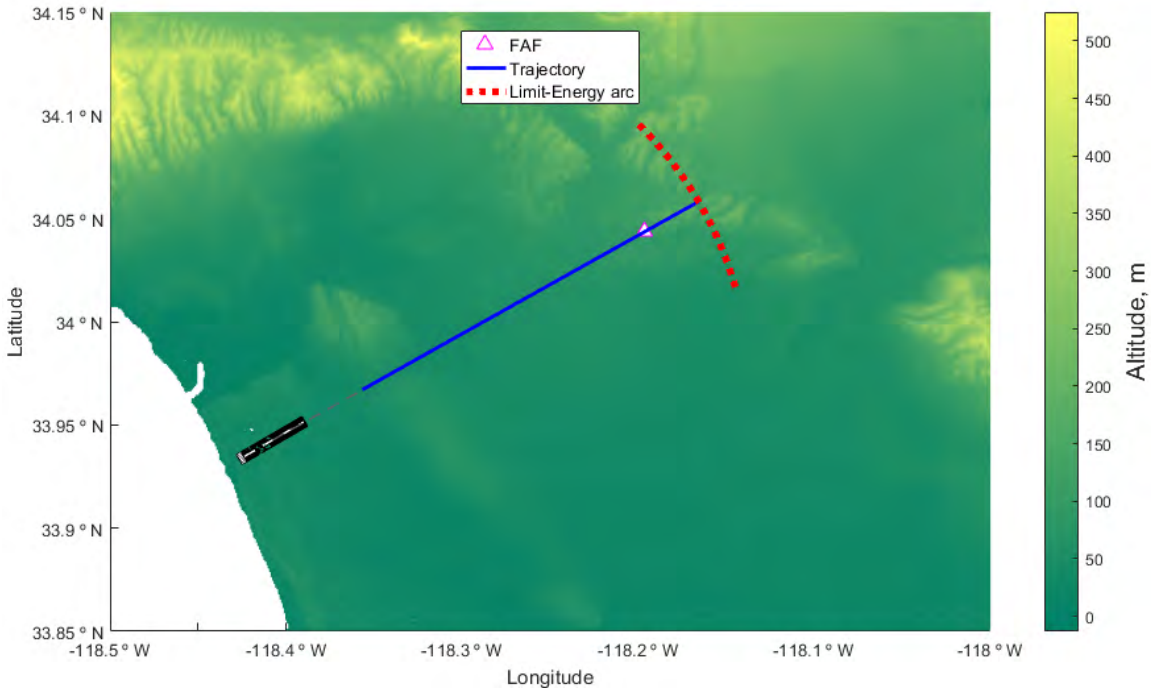


Figure 6.10: Energy-arc (red-dashed arc) seen from the top. For a given aircraft altitude and speed, a stabilized approach is possible as long as the aircraft is beyond the arc.

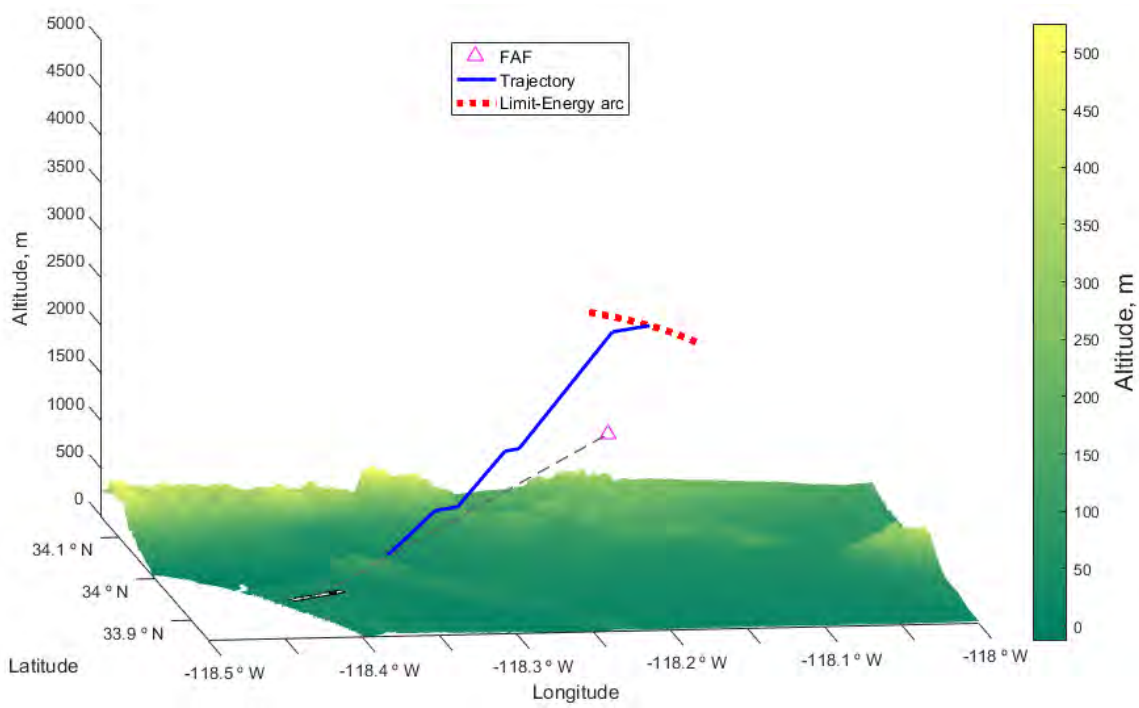


Figure 6.11: Energy-arc in a 3-D environment. The altitude profile shown in blue represents the last trajectory that stabilizes the aircraft. As soon as the aircraft traverses the energy-arc, the approach is non-stabilized.

6.2.4 Conclusion

The implementation of the energy-limit trajectory provides a complementary information to flight crews and improves the energy awareness, with the aim of reducing the number of non-stabilized approaches and, ultimately, go-around procedures. The function provides flight crews with additional information, in the form of the energy-limit trajectory, in order to improve their energy awareness on a real-time basis. The function is useful when the aircraft has deviated from its intended lateral path and, then, flight crews shall estimate the minimum distance required to land for the current aircraft state. In particular, two situations are likely to occur:

- Stabilized approach possible: the aircraft is beyond the energy-circle, the energy arc indicates the proximity of the limit trajectory that ensures stabilization. Then, flight crews may communicate the information with ATC to avoid going around.
- Stabilized approach not possible: the aircraft is within the energy-circle, i.e. the aircraft distance to the runway is lower than the distance provided by the energy-limit trajectory. In that case, the stabilized approach is not possible and pilots can communicate with ATC in order to be rescheduled in the arrival sequence. The energy-limit trajectory provides the minimum path distance to stretch.

The provision of the energy-limit trajectory information reduces both ATC and crews workload, since go-around maneuvers are avoided and aircraft can be rescheduled in the arrival sequence at an early stage. The cost of going around is minimized because, in this case, the aircraft only needs to be reinserted in the arrival flow. The energy-circle concept constitutes a simple visualization aid displayed on the ND, which provides awareness of the proximity of the energy-limit ground distance. As of today, the energy-circle is only a proposal, this information could be complemented with the visualization on the Vertical Display (VD) or, alternatively, an EFB can host the function.

The preliminary operational assessment performed in the flight simulator suggested that the proposed energy-limit trajectory represents the physical limit but it is not flyable from an operational point of view. Therefore, further operational assumptions improve the follow up of the energy-limit trajectory using current aircraft guidance laws. Nonetheless, further evaluation with flight test pilots and system experts could provide more assumptions in light of improving the operational representativeness of these trajectories before flight tests. The real-time computation and visualization of the energy-limit trajectory is another key point that has to be addressed. In conclusion, three situations have been described in this thesis. In the first one, aircraft are in cruise or descent and the algorithm provides the optimal profile which reaches the aircraft position. In the second, aircraft

are in high-energy condition in approach and the algorithm computes the trajectory that stabilizes the aircraft. In the third scenario, no safe solution exists and the algorithm explores the whole search space. The provision of the energy-limit trajectory avoids to explore the whole search space and provides the minimal distance-to-stretch which ensures stabilization. The next chapter summarizes the contributions of this thesis, and suggests several directions for improving the maturity of the present algorithm.

7

Conclusion

First, this chapter summarizes the benefits of the solution proposed in the thesis. Secondly, the enablers for the function and flight operations perspectives are described. Finally, several directions for future work related to trajectory optimization of descent and approach paths are presented.

7.1 Achievements

The initial objective of this thesis was to provide improvements to current FMS design with regard to the calculation of descent and approach paths. In those phases, when aircraft are off the path defined in the flight plan, the trajectory calculated by the system is based on hypotheses that do not update according to the dynamic conditions of flight. This objective has been completely fulfilled, since the methodology proposed in this thesis, which is based on an A* algorithm, computes vertical fuel-efficient permanent trajectories for any aircraft energy condition. The term permanent refers to the continuous availability of the trajectory that reaches aircraft position, regardless of the present modes and energy condition.

The algorithm has been generalized for solving any arrival procedure of any airport in the world. This is a strong contribution, since most works in the literature focus on standard arrival procedures, and then they can not be extrapolated to any other airspace design. Moreover, the computed trajectories rely on enhanced energy management where aircraft no longer follow conventional Mach/CAS trajectories but energy-efficient trajectories. These improvements increase the degree of freedom of the aircraft, which is free to choose the optimal speed at each flight segment rather than following the optimal speed defined by means of the CI. Hence, vertical discontinuities in the flight plan are likely to be solved by simply applying the correct energy management strategy. In terms of optimality, these trajectories are more adapted to the particular airspace design, since they are constructed on the basis of optimal flight strategy instead of fixed hypotheses. Besides, the computation of the energy-limit trajectory on a real-time basis complements the algorithm in those cases where no solution exists. The trajectory is continuously provided to flight crews, improving energy awareness with the purpose of reducing non-stabilized approaches and go-around procedures. The thesis contributes to the automatic trajectory generation concept, which is part of more efficient avionic systems and may pave the way for next generation of FMSs.

7.2 Enablers for this type of flight operation

The algorithm presented in this thesis computes fuel-efficient permanent trajectories, which represent a new operational concept as they are based on a type of operation which differs from nowadays standard operations. Therefore, a set of assumptions needs to be addressed before putting in place this type of operations:

- Reliance on the lateral path: The vertical trajectories computed in this thesis assume

that the lateral path is known. The distance to the destination links the lateral and the vertical path. The algorithm computes the optimal trajectory for the given distance to the destination provided by the lateral path.

- Improvements on ground segment: Predictability is a key element in ATC. Improvements on trajectory share could enable this type of operation, since ATCOs are aware of aircraft intentions. The concept should be in line with ATCOs ways of working and operational refinements such as definition of speed corridors and inclusion of time window constraints may be required.

- New guidance modes: As of today, flight crew actions are required to follow the computed trajectories, which represents an increase of workload. The definition of novel energy-based flight modes to automatically guide aircraft along the trajectory would drastically decrease this workload. In order to exploit the full potential, the extension of secondary flight controls such as landing gear, flaps and airbrakes shall be automated, since path-following requires accuracy and fine control to minimize energy deviations.

7.3 Perspectives and future work

7.3.1 Flight operations perspective

The tests performed in Airbus flight simulators confirmed the physical flyability of the computed trajectories using current guidance laws. This preliminary assessment suggests that tracking such trajectories is feasible without strong modification of current flight control laws. However, the increase of workload that these continuous changes of FPA induces to flight crews, is likely to be unacceptable. The computed trajectories could be manually followed by flight crews, but the automatic tracking of the trajectory would be always better in terms of accuracy. The interest of the permanent trajectory is the continuous availability of a trajectory that can be followed in managed modes when aircraft are deviated from their initial flight plans, and then the definition of guidance laws for that purpose is needed.

The operational concept proposed in this thesis should be reviewed with flight test pilots in order to improve its maturity. Probably, additional constraints should be added in compliance with current standards and flight operations. The trajectory information sharing is crucial for this type of operations so that it may be interesting to interface with ATCOs, since the concept may be seen as a step forward on current CDO. Airspace designs are clearly evolving, for instance, novel procedures in the US usually contain an

altitude corridor. In a similar way, speed corridors could be defined, in which the aircraft speed shall remain at any moment. The combination of these speed corridors with time window constraints would improve the predictability. In that context, ATCOs may provide clearances by blocks, within which flight crews manage the energy of the aircraft according to airlines policy.

The test campaign should be resumed in order to evaluate the algorithm in other scenarios under different wind and ambient conditions. Eventually, once the maturation phase finishes, the concept could be tested in flight. Finally, the deterministic nature of the algorithm makes certification possible. The influence of discretization may be assessed in detail to minimize its impact in the final solution. A possible solution is to apply path smoothing techniques once the computation is finalized. For instance, the algorithm may be hosted in an EFB connected to the avionics suite. The initial computation of the trajectory is performed in the EFB and sent to the FMS, which performs a calculation using conventional integrators on the basis of the input trajectory. Then, regular exchanges between both systems ensure the trajectory is updated according to the conditions of flight.

7.3.2 Model improvements

As it was discussed during the presentation of high-energy scenarios in chapter 5, improvements on airbrakes models are required to measure their impact on aircraft structure and fatigue cycles. The lack of differentiation between nodes generated using airbrakes and those without using airbrakes forces the algorithm to explore more nodes. A cost degradation of those nodes generated using airbrakes is not the appropriate solution because, if the airspace design requires the use of airbrakes to satisfy some constraint, the algorithm may explore all nodes generated without airbrakes before those generated with airbrakes, which results in large computation times. Similarly, cost improvements can also be performed. Nowadays, expenses for airlines are synthesized by the CI, which represents the ratio of time and fuel costs of a given aircraft. The objective function to minimize is the sum of fuel and time costs, the latter is converted into units of kilograms through the CI. Although the generated trajectories are (global) optimal, the optimization is performed aircraft by aircraft, and then it may not be optimal from a network perspective. For instance, a higher cost for aircraft x could mean large savings for aircraft y, this solution being more optimal than optimizing x and y separately. Hence, improvements on the representation of airlines costs may build meaningful objective functions yielding bigger savings to airlines.

7.3.3 Heuristics using Artificial Neural Networks

The definition of a meaningful heuristic function for highly non-linear problems is a difficult task. While sometimes knowledge of the problem is sufficient for defining a good heuristic, in other situations this is not enough to classify nodes properly. One direction of research proposed is to use neural networks as the evaluation function (see Fig.7.1). For this purpose, a patent has been filed during the thesis [110]. The basic idea is that, based on the initial and the final state (L_1), the algorithm estimates the cost of going from the initial state to the final state (L_2). The training data of the neural network could be the compilation of point-to-point constraint-less optimal trajectories. Constraints shall be disregarded as they degrade the optimal solution so that the admissibility property is always satisfied. The neural network-based heuristic function gives an estimate of the optimal cost between states A and B. In the real problem, if there is no constraint between A and B, the estimate would be exactly the optimal cost from A to B. If there is an optimal constraint, the estimate would be optimistic, which means the estimate cost is lower than the actual optimal cost due to the presence of a constraint. Weather conditions shall be taken into account as they impact aircraft performance and may turn an admissible heuristic into a non-admissible one. For instance, if the optimization criterion is fuel, the neural network yields an estimation of the minimum fuel to go from node A to node B.

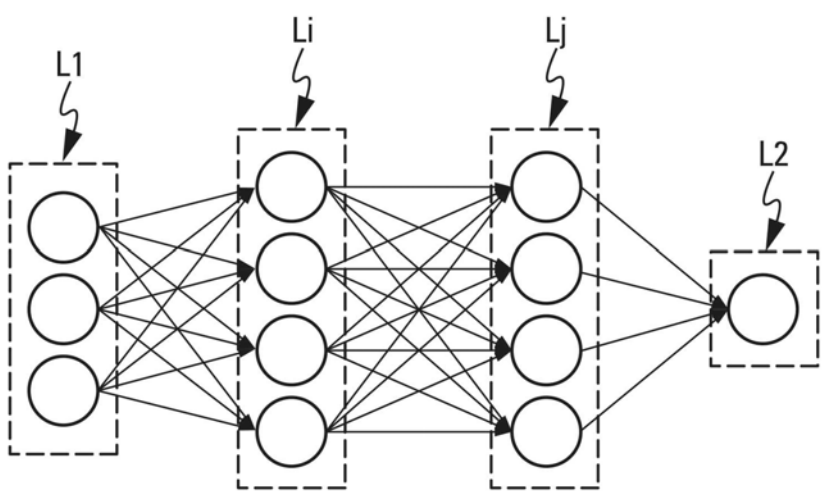


Figure 7.1: Schema of an artificial neural network.

7.3.4 From the conception towards the industrial application

The algorithm proposed in this thesis demonstrates that several improvements on trajectory generation could bring fuel savings. The continuity of the trajectory (permanent)

makes that the optimal flight strategy is continuously available to flight crews, since readjustments are done according to the dynamic conditions of flight. As of today, the design of the algorithm does not take into account this notion of real-time calculations. In reality, the algorithm is relaunched for different final conditions and the search space is generated again. One of the directions of work would be to interface the algorithm with other systems, so that it processes all input information on real-time. Then, the definition of thresholds that define the boundary between tactic adjustments, which could be simply performed through guidance laws, and strategic revisions, where the complete trajectory is recomputed as the previous one is considered obsolete. As to the strategic revisions, the algorithm may reuse most nodes of the search space generated at the previous calculation. Path costs of each of the nodes are maintained while heuristics costs should be updated. As a consequence, the A* explores a preprocessed search space. Probably, this approach would result in a first run relatively expensive, but next adjustments would be cheap. In a similar way, the algorithm could consider hierarchical planning and landmark definition, although attention shall be put on the admissibility of the heuristic, which requires further investigation. This concepts are also applicable to the generation of energy-limit trajectories.

7.4 Wrap-up

This thesis puts in evidence the advantages of path-finding algorithms to find global optima in graphs constructed incrementally. The number of constraints that apply to trajectory optimization problems, especially during descent and approach phases, reduces the combinatorial problem. The use of heuristic functions constructed from known information of the problem also helps to converge to the solution. While real-time computations still present a challenge that needs to be addressed, this thesis establishes a promising baseline to generate fuel-efficient permanent aircraft trajectories which brings significant improvements to flight operations efficiency and safety.

A

Appendix A: Reference frames and equations of motion

A.1 Reference frames and equations of motion

Aircraft motion in the vertical plane is subjected to several forces and accelerations. To relate them, three reference axis are defined: earth (x_e, z_e), body (x_b, z_b) and aerodynamic (x_a, z_a) as those displayed in Fig. A.1:

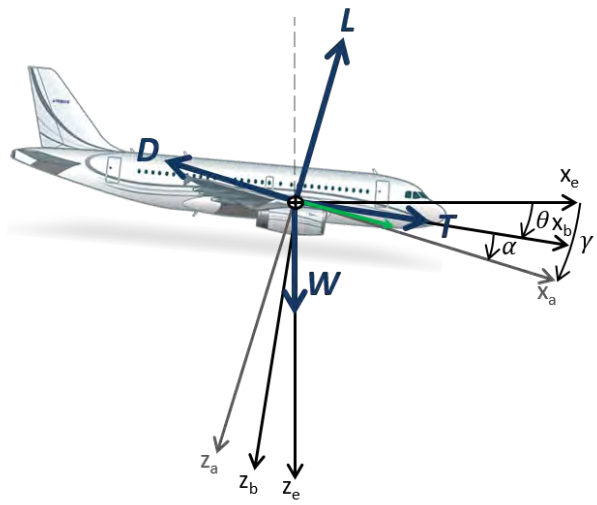


Figure A.1: Forces applied on aircraft during descent.

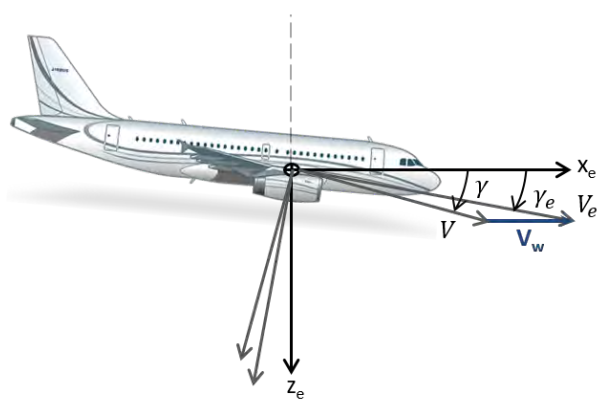


Figure A.2: Influence of wind in flight path.

The application of Newton's second law to a variable mass system is represented by the following equation:

$$\sum F_{ext} + V_{rel}\frac{dm}{dt} = m\frac{dV}{dt} \tag{A.1}$$

Where F_{ext} represents the external forces applied to the body and V_{rel} is the relative velocity of the incoming mass. Considering that the mass variation is very small compared to net forces, such that $\frac{dm}{dt} \ll F_{ext}$, Eq.(A.1) can be rewritten as:

$$\sum F_{ext} = m\frac{dV}{dt} \tag{A.2}$$

The relation between aircraft external forces defined in the aerodynamic frame (x_a, z_a) and a fixed Galilean frame (x_e, z_e) in the presence of wind (V_w), as shown in Fig. A.2, is given by the following expression:

$$\dot{x_e} = V\cos\gamma + V_w(s,z) = V_e\cos\gamma_e \tag{A.3}$$

$$\dot{z_e} = V\sin\gamma = V_e\sin\gamma_e \tag{A.4}$$

The application of Newton's second law for the aircraft motion results in the earth frame is given by:

$$\sum F_x = (Thr - D)\cos\gamma - L\sin\gamma \tag{A.5}$$

$$\sum F_z = L\cos\gamma + (Thr\cos\alpha - D)\sin\gamma - mg_0 \tag{A.6}$$

In previous Eqs.(A.5) and (A.6), the angle of attack α has been neglected, so the thrust is collinear with the velocity vector. This assumption is valid because $Thr\cos\alpha \approx Thr$ and the vertical component of thrust is very small compared to the lift force so that $Thr\sin\gamma << L$. Then, accelerations are described as the sum of aircraft and wind accelerations:

$$a_x = \dot{V}\cos\gamma - V\dot{\gamma}\sin\gamma + \frac{\partial V_w(s,z)}{\partial s}\dot{s}_e(t) + \frac{\partial V_w(s,z)}{\partial z}\dot{z}_e(t)) \tag{A.7}$$

$$a_z = \dot{V}\sin\gamma + V\dot{\gamma}\cos\gamma \tag{A.8}$$

It has to be noted that wind only applies to the horizontal plane $(V_w = V_{w_s} \rightarrow V_{w_z} = 0)$. The combination of Eqs.(A.5) and (A.6) with Eqs.(A.7) and (A.8) gives the following expression:

$$\sum F_x = ma_x \Rightarrow \quad m(\dot{V}\cos\gamma - V\dot{\gamma}\sin\gamma + \frac{\partial V_w(s,z)}{\partial s}\dot{s}_e(t) + \frac{\partial V_w(s,z)}{\partial z}\dot{z}_e(t)) = (Thr\cos\alpha - D)\cos\gamma - L\sin\gamma \tag{A.9}$$

$$\sum F_z = ma_z \Rightarrow \quad m(\dot{V}\sin\gamma + V\dot{\gamma}\cos\gamma) = (Thr\sin\alpha + D)\sin\gamma + L\cos\gamma - mg_0 \tag{A.10}$$

In order to relate earth (x_e, z_e) and aerodynamic (x_a, z_a) reference frames, a rotation matrix is used:

$$R(\gamma) = \begin{bmatrix} \cos\gamma & \sin\gamma \\ -\sin\gamma & \cos\gamma \end{bmatrix} \tag{A.11}$$

The projection of (A.9) and (A.10) on the aerodynamic reference frame (x_a, z_a) using the rotatin matrix defined in Eq.(A.11), yields:

$$m(\dot{V}+\cos\gamma\frac{\partial V_w(s,z)}{\partial x}(V\cos\gamma+V_w(s,z))+\cos\gamma\frac{\partial V_w(s,z)}{\partial z}V\sin\gamma) = (Thr\cos\alpha-D)-mg_0\sin\gamma) \tag{A.12}$$

$$m(V\dot{\gamma} - \sin\gamma \frac{\partial V_w(s,z)}{\partial x}(V\cos\gamma + V_w(s,z)) - \sin^2\gamma \frac{\partial V_w(s,z)}{\partial z} V) = Thr\sin\alpha + L - mg_0\cos\gamma \tag{A.13}$$

A further simplification is made since wind gradients are disregarded for trajectory generation purposes. This assumption yields the following formulation of the equations of motion:

$$\dot{s} = V\cos\gamma + V_w \tag{A.14}$$

$$\dot{z} = V\sin\gamma \tag{A.15}$$

$$\dot{V} = \frac{Thr - D - mg_0\sin\gamma}{m} \tag{A.16}$$

$$\dot{\gamma} = \frac{L - mg_0\cos\gamma}{mV} \tag{A.17}$$

$$\dot{m} = \frac{dm}{dt} = -FF \tag{A.18}$$

Where Eqs.(A.14) and (A.15) represent the kinematics of the aircraft, Eqs.(A.16) and (A.17) the dynamics and the final Eq.(A.18), models the variation of mass as a function of time. In general, slow dynamics are dominant in trajectory generation processes whereas fast dynamics are specially relevant in the definition of flight control laws. This is the reason why most trajectory generation problems disregard the dynamics of the FPA, such that $\dot{\gamma} = 0$. Flight control laws are embedded in specific flight control computers, whereas FM computers provide the trajectory for the flight plan. A classical control laws for this type of application could be defined as:

$$\dot{\gamma} = K(\gamma_c - \gamma) \tag{A.19}$$

Where K is a tunable gain, γ_c is the commanded (aerodynamic) FPA and γ is the current FPA. The behavior of the control law depends on the value of K. Therefore, this assumption reduces aircraft equations of motion to the following formulation:

$$\begin{cases} \dot{s} = \dfrac{ds}{dt} = V\cos\gamma_c + V_w \\ \dot{z} = \dfrac{dz}{dt} = V\sin\gamma_c \\ \dot{V} = \dfrac{dV}{dt} = \dfrac{Thr - D - mg_0\sin\gamma_c}{m} \\ \dot{m} = \dfrac{dm}{dt} = -FF \end{cases} \tag{A.20}$$

B

Appendix B: Atmospheric Model

B.1 The atmospheric model

The atmospheric model used in the algorithm is described in [91], which is based on the International Standard Atmosphere (ISA) model published by the International Civil Aviation Organization [111]. The BADA model describes that ambient temperature decreases as altitude increases below the tropopause, which delimits the frontier between the troposphere and stratosphere:

$$T = T_0 + \Delta\text{ISA} + \beta_T \cdot h \tag{B.1}$$

Where T is the Total Air Temperature (TAT), T_0 is the standard atmospheric temperature at mean sea level, Δ_{ISA} is the temperature variation with respect to standard atmosphere conditions, β_T is the ISA temperature gradient below the tropopause and h is the aircraft altitude. Above the tropopause, air temperature for commercial aviation is considered to be constant:

$$T = T_0 + \Delta\text{ISA} + \beta_T \cdot h_{tropo} \tag{B.2}$$

Where h_{tropo} is the tropopause altitude defined at 11000 m. Air pressure p decreases with altitude:

$$p = p_0 + \left(\frac{T - \Delta\text{ISA}}{T_0}\right)^{-\frac{g_0}{\beta_T R}} \tag{B.3}$$

Where p_0 is the standard atmospheric pressure at mean sea level and R the real gas constant for air. All altitudes above the tropopause, the following expression is applied:

$$p = p_{tropo} \cdot \exp\left(-\frac{g_0}{\beta_T R}(h - h_{tropo})\right) \tag{B.4}$$

Consequently, air density (ρ) decreases as altitude increases:

$$\rho = \frac{p}{RT} \tag{B.5}$$

The speed of sound a decreases with ambient temperature as follows:

$$a = \sqrt{\kappa R T} \tag{B.6}$$

Where κ is the adiabatic index of air. The conversions between calibrated airspeed V_{CAS} and true airspeed V_{TAS}, and vice-versa, are performed by Eqs.(B.7) and (B.8):

$$V_{TAS} = \left[\frac{2p}{\mu\rho}\left\{\left(1 + \frac{p_0}{p}\left[\left(1 + \frac{\mu\rho_0}{2p_0}{V_{CAS}}^2\right)^{\frac{1}{\mu}} - 1\right]\right)^{\mu} - 1\right\}\right]^{\frac{1}{2}} \tag{B.7}$$

$$V_{CAS} = \left[\frac{2p_0}{\mu\rho_0} \left\{ \left(1 + \frac{p}{p_0} \left[\left(1 + \frac{\mu\rho}{2p} {V_{TAS}}^2 \right)^{\frac{1}{\mu}} - 1 \right] \right)^{\mu} - 1 \right\} \right]^{\frac{1}{2}} \tag{B.8}$$

Where ρ is the standard atmospheric density at sea level and the term μ is defined by:

$$\mu = \frac{\kappa - 1}{\kappa} \tag{B.9}$$

Finally V_{TAS} can be converted into the dimensionless Mach number (M) through Eq. (B.10):

$$M = \frac{V}{a} \tag{B.10}$$

C

Appendix C: Gas Emissions Model

C.1 Gas emission model

The model used in this thesis for the estimation of aircraft emissions is based on the Advanced Emission Model 3 (AEM3) proposed by Eurocontrol [112], which proposes some corrections to the Boeing Method 2 (BM2) [113]. The first step is to compute the pressure (δ) and the temperature (θ) ratio for ambient conditions:

$$\delta = \frac{p}{p_0} \tag{C.1}$$

$$\theta = \frac{T}{T_0} \tag{C.2}$$

The actual fuel flow value (FF), which is computed for a certain segment of the trajectory, needs to be corrected, since International Civil Aviation Organization (ICAO) engine data have been collected from unmounted engines on test benches under ideal atmospheric conditions:

$$W_{ff} = \frac{FF}{\delta}\theta^{3.8} \cdot \exp(0.2M^2) \tag{C.3}$$

Where W_{ff} is the corrected fuel flow value. The humidity factor H is computed as:

$$H = -19.0 \cdot (\omega - 0.0063) \tag{C.4}$$

Where the specific humidity ω factor is defined by:

$$\omega = \frac{0.62198(\Phi)p_v}{p - 0.37802(\Phi)p_v} \tag{C.5}$$

In Eq. (C.5), terms p_v and Φ represent the saturation vapor pressure and the relative humidity respectively. The corrected fuel flow (W_{ff}) value computed in Eq. (C.3) is interpolated with reference data obtained from ICAO data lake in order to obtain the correspondent reference Emission Indices (EI) (REI), which combined with the ambient conditions, yields the actual EI:

$$\begin{cases} \text{EIHC} = \text{REIHC} \cdot \theta^{3.3}/\delta^{1.02} \\ \text{EICO} = \text{REICO} \cdot \theta^{3.3}/\delta^{1.02} \\ \text{EINO}_x = \text{REINO}_x \cdot (\theta^{3.3}/\delta^{1.02})^{0.5} \cdot \exp(H) \end{cases} \tag{C.6}$$

The gas emissions studied are: hydrocarbon (HC), carbon monoxide (CO), nitrogen oxide (NO_x), water (H_2O), carbon dioxide (CO_2) and sulfur oxides (SO_x). The EI for

H_2O, CO_2 and SO_x are constant values expressed in g/kg:

$$\begin{cases} \mathrm{EIH_2O} = 1237 \\ \mathrm{EICO_2} = 3155 \\ \mathrm{EISO}_x = 0.8 \end{cases} \tag{C.7}$$

It can be observed that their value is proportional to fuel flow. Finally, gas emissions in kilograms are computed by means of the following expression:

$$\text{Gas emissions} = \text{n}^{\text{o}} \text{ of engines} \cdot \sum_{i=1}^{N} (\mathrm{EIHC,EICO,EINO}_x ...) \cdot FF \cdot dt \cdot 10^{-3} \tag{C.8}$$

Where N equals to the number of segments of a given trajectory and dt is the time lapse of a single segment.

Bibliography

[1] Mulder M. van Passen M. M. Amelink, M. H. J. and J. Flach. Theoretical foundations for a total energy-based perspective flight-path display,. *The International Journal of Aviation Psychology, Vol. 15, No. 3 pp. 205–231.*, July 2005.

[2] Eurocontrol. Long-Term Forecast: IFR Flight Movements 2010-2030. Long-term forecast: Ifr flight movements 2010-2030. *Rep. CND/STATFOR Doc*, 415, 2010.

[3] Airbus. Global market forecast 2016-2035 - mapping demand. 2016.

[4] Eurocontrol. European aviation in 2040. challenges of growth. *Annex 4, Network Congestion*, 2018.

[5] EASA. *Continued Growth of Aviation poses Environmental Challenges*. European Aviation Safety Agency, 2019.

[6] SESAR European ATM Master Plan. Excecutive view. Technical report, Publications Office of European Union, Edition SESAR JU & SESAR Work Package C and Partners, 2015.

[7] NextGen Office. Faa's nextgen implementation plan 2016. *Tech. rep*, 800, 2016.

[8] Kevin R. Driscoll. Cyber safety and security for reduced crew operations (rco). *Honeywell International, Inc.*, page 27, 2016.

[9] European Comission. Clean sky 2: developing new generations of greener aircraft.

[10] ICAO. *Manual on System Wide Information Management (SWIM) concept.*

[11] SESAR Joint Undertaking, editor. *Trajectory Based Operations (TBO)*, ATC Global, Beijing, 12 September 2016.

[12] SESAR Joint Undertaking. Extended projected profile (epp) availability on ground. `https://www.sesarju.eu/sesar-solutions/enabling-aviation-infrastructure/extended-projected-profile-epp-availability-ground`.

[13] Nelson M. Guerreiro and Matthew C. Underwood. Understanding extended projected profile (epp) trajectory error using a medium-fidelity aircraft simulation. *AIAA Aviation Forum*, June 2018.

[14] EASA. *Annual Safety Review*. 2017.

[15] Defense Advanced Research Projects Agency (DARPA). A DARPA perspective on artificial intelligence.

[16] R. Walter. Flight management systems, 2001.

[17] S. Liden. The evolution of flight management systems. *Honeywell Air Tramport Systems, PO Box*, 21(11):7803–24254, 1994.

[18] D. Avery. *The Evolution of Flight Management Systems*. February, 2011.

[19] Airbus. Statistical analysis of commercial aviation accidents 1958-2015 - airbus. 3170, 2016.

[20] D. Manzey S. Müller, K.Schreiter; R. Luckner. Development, design, and flight test evaluation of a continuous descent approach procedure for nighttime operation at louisville international airport. *Report No. Partner-COE-2005-002*, January, 2006.

[21] Airbus. Descent and approach profile management, Oct 2006.

[22] Airbus. Aircraft energy management during approach, Oct 2005.

[23] Airbus. Flying stabilized approaches, Oct. 2006.

[24] Airbus. Human factors aspects in incidents / accidents, May 2004.

[25] Airbus. Operations golden rules, Jan. 2004.

[26] Anil V. Rao. A survey of numerical methods for optimal control. *AAS 0*, pages 9–334.

[27] R.W.H. Sargent. Optimal control. *Journal of Computational and Applied Mathematics 124 (2000) 361371*, 1999.

[28] A. E. Bryson and Y. Ho. *Applied Optimal Control*. Wiley, 1975.

[29] D. Delahaye, S. Puechmorel, P. Tsiotras, and E. Feron. Mathematical models for aircraft trajectory design : A survey. *Lecture notes in Electrical Engineering Lecture Notes in Electrical Engineering*, 10(1007):431–54475, 2014.

[30] A. I. Tyatyushkin A. Yu. Gornov and E. A. Finkelstein. Institute of system dynamics and control theory. *Russian Academy of Sciences, ul. Lermontova 134, Irkutsk, 664033 Russia*, 2012.

[31] O. yon Stryrk and R. Bulirsch. Direct and indirect methods for trajectory optimization. *Annals of Operations Research 37(1992)357-373*, Mathematisches Institut, Technische Universitat Munchen, Postfach 20 24 20, D-8000 Munchen 2, Germany.

[32] L. S. Pontryagin, V. G. Boltyanski, R. V. Gamkrelidze, and E. F. Mischenko. The mathematical theory of optimal processes, new york, ny: Interscience publishers. 1962.

[33] D. Rivas A. Franco, A. Valenzuela. Optimal control of cruise flight at constant altitude. *ICAS2010 - 27th International Congress of the Aeronautical Sciences*, 2010.

[34] J. Villarroel and L. Rodrigues. *An Optimal Control Framework for the Climb and Descent Economy Modes of Flight Management Systems*. Concordia University Montreal, Canada.

[35] S.L. Kek. Nonlinear programming approach for optimal control problems. *Proceedings The 2nd International Conference On Global Optimization and Its Applications 2013 (ICoGOIA2013) Avillion Legacy Melaka Hotel, Malaysia.*

[36] C. R. Hargraves and S. W. Paris. Direct trajectory optimization using nonlinear programming and collocation. *Journal of Guidance, Control and Dynamics*, 10(4):338–342, 1987.

[37] J. T. Betts and E. J. Cramer. Application of direct transcription to commercial aircraft trajectory optimization. *Journal of Guidance, Control and Dynamics*, 18(1):151–159, 1995.

[38] R. Fernandes and C. Beskens. Benefits of optimal flight planning on noise and emissions abatement at the frankfurt airport. *AIAA Guidance, Navigation and Control Conference*, August 2012.

[39] X. Liu C. Li and J. Shen Q. Ma. Four-dimensional descent trajectory optimization for aircraft in turbulence. *Proceedings of the IEEE International Conference on Information and Automation Ningbo.*

[40]

[41] P. Singla R. Bhattacharya. Nonlinear trajectory generation using global local approximations. *Proceedings of the 45th IEEE Conference on Decision & Control*, Manchester Grand Hyatt Hotel, San Diego, CA, USA, December 13-15, 2006.

[42] Y. Zhao. Efficient and robust aircraft landing trajectory optimization. *Ph.D. thesis, School of Aerospace Engineering, Georgia Institute of Technology*, May 2011.

[43] Y. Zhao and P. Tsiotras. Density functions for mesh refinement in numerical optimal control. *Journal of Guidance, Control, and Dynamics*, 1:271, 2010.

[44] Bellman R. Dynamic Programming. 4th edition, princeton university press. 1965.

[45] D. P. Bertsekas. *Dynamic Programming and Optimal Control*, volume Volume 1, Third Edition. Athena Scientific, Belmont, Massachusetts, 2005.

[46] N. Berend C. Talbot M. Haddou J. Laurent-Varin, F. Bonans. On the refinement of discretization for optimal control problems. *16th IFAC Symposium on Automatic Control in Aerospace*, 2004.

[47] Mihaela Botez R. Felix Patrón, R.S. Aircraft flight trajectories optimization through genetic algorithms for a LNAV and VNAV integrated path. *Journal of Aerospace Information Systems*, 2015.

[48] A. Bar-Gill E. Rippel and N. Shimkin. Fast graph-search algorithms for general aviation flight trajectory generation. *Technion—Israel Institute of Technology, Haifa, Israel.*

[49] J. Vasalek J. Doebbler, P. Gesting. Real-time path planning and terrain obstacle avoidance for general aviation aircraft. *AIAA Guidance, Navigation, and Control Conference and Exhibit. AIAA 2005-5825.*

[50] R. Devulapalli. An efficient algorithm for commercial aircraft trajectory optimization in the air traffic system.

[51] Tasos Nikoleris, Gano B. Chatterji, and Richard A. Coppenbarger. Comparison of fuel consumption of descent trajectories under arrival metering. *Journal of Aircraft*, 53:6, 2016.

[52] M. Soler. *Commercial Aircraft Trajectory Planning based on Multiphase Mixed-Integer Optimal Control*. Universidad Juan Carlos III de Madrid. Omm Editorial, 2013.

[53] M. Soler, D. Zapata, A. Olivares, E. Staffetti, and J. Cegarra. Comparative analysis of commercial aircraft trajectory performance. 2nd international conference on engineering optimization. *September*, pages 6–9, Lisbon, Portugal 2010.

[54] J. T. Betts. Survey of numerical methods for trajectory optimization. *Journal of Guidance, Control, and Dynamics*, 21(2):193–207, 1998.

[55] S. Franco, D. Rivas, and A. Valenzuela. Minimum-fuel cruise at constant altitude with fixed arrival time. *Journal of Guidance, Control and Dynamics*, 33(1):280–285, 2010.

[56] Pérez-Batlle M. Dalmau, R. and Prats X. Real-time identification of guidance modes in aircraft descents using surveillace data. *37th Digital Avionics System Conference*, 2018.

[57] H.G. Visser. A 4-D trajectory optimization and guidance technique for terminal area traffic management, June 1994.

[58] R. Subramanian, S. Roberto, G. Alessandro, and L. Yifang. *Novel Flight Management System for Real-Time 4-Dimensional Trajectory Based Operations*. in AIAA Guidance, Navigation, and Control (GNC) Conference American Institute of Aeronautics and Astronautics, 2013.

[59] Kevin R. Bruce. Nasa B737 flight test results of the total energy control system. *(NASA-CR-778285). N87-26918. Boeing Commercial Airplane Company*, Jan 1987.

[60] H.Erzberger. Automation of on-board flight path management. *(NASA-TM- 84212) (NASA) 22 P CSCL 07C HC A02/MF A01. N82-16088*, December, 1981.

[61] M. Lefebvre. Gestion automatisée de l'energie d'un avion de transport civil: Application aux phases de descente et d'approche. *Doctorat Université de Toulouse - Paul Sabatier III*, 2012.

[62] P. M. A. De Jong, N. De Gelder, F. J. L. Bussink, R. P. M. Verhoeven, R. Kohrs, and M. Mulder. Time and energy management during descent and approach for aircraft: A batch-simulation study. *Journal of Aircraft*, 2013.

[63] N. de Gelder R. Dalmau, R. Verhoeven and X. Prats. Performance comparison between temo and a typical fms in presence of cta and wind uncertainties. *978-1-5090-2523-7/16 - IEEE*, 2016.

[64] Sam G. Park and John Paul Clarke. Vertical trajectory optimization for continuous descent arrival procedure. *AIAA Guidance, Navigation, and Control Conference, Minneapolis, Minnesota*, August 2012.

[65] Sam G. Park and John Paul Clarke. Vertical trajectory optimization to minimize environmental impact in the presence of wind. August 2015.

[66] P. Hagelauer and F. A Mora-Camino. soft dynamic programming approach for online aircraft 4d trajectory optimization. *European Journal of Operational Research*, Vol. 10(1):87–95, 1998.

[67] Y. Miyazawa N. K. Wickramasinghe, A. Harada. Flight trajectory optimization for an efficient air transportation system. *28th International Congress of the Aeronautical Sciences. ICAS*, 2012.

[68] L. De Filippis and G. Guglieri. Advanced graph search algorithms for path planning of flight vehicles. *Recent Advances in Aircraft Technology*, February 2012.

[69] Fethi Abdelmoula and Marco Scholz. Lnas - a pilot assistance system for low-noise approaches with minimal fuel consumption. *31st Congress of the International Council of the Aeornautical Sciences. Belo Horizonte, Brasil*, 9-14 September, 2018.

[70] Eugene M. Cliff. Energy-state models. Appril 23, 1998.

[71] D. Manzey S. Müller, K.Schreiter; R. Luckner. Manual flying and energy awareness. *Aviation Psychology and Applied Human Factors. 7(1), 18–27. https://doi.org/10.1027/2192-0923/a000111*, 2017.

[72] D. G. Hull. *Fundamentals of Airplane Flight Mechanics.* Springer, Berlin, Heidelberg, Germany, 2007.

[73] X. Wang. Solving optimal control problems with MATLAB - indirect methods.

[74] Bertram Raphael Peter E. Hart, Nils J.Nilsson. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions of systems science and cybernetics, Vol.SSC-4,NO.2*, July 1968.

[75] J. A. Bondy and U.S.R Murty. *Graph Theory.* Springer-Verlag London, 2008.

[76] R. Diestel. *Graph Theory.* Springer-Verlag New York, 2017.

[77] Amit's A* pages. `http://theory.stanford.edu/~amitp/GameProgramming/`. Accessed: 2010-09-30.

[78] Andrew V. Goldberg and Chris Harrelson. Computing the shortest path: A* search meets graph theory. *Technical Report MSR-TR-2004-4*, 2014.

[79] Richard E. Korf. Real-time heuristic search. *Arriljcial Intelligence 42 189-211*, 1990.

[80] S. Koenig and M. Likhachev. Real-time adaptive A*. *AAMAS 2006 May 812, Hakodate, Japan.*, 2006.

[81] M. Likhachev S. Koenig and D. Furcy. Lifelong planning A*. *Elsevier Science*, 2005.

[82] A. Stentz and M. Hebert. A complete navigation system for goal acquisition in unknown environments. *In Autonomous Robots, Volume 2, Number 2, August*, 1995.

[83] D. Harabor and A. Grastien. Online graph pruning for pathfinding on grid maps. *Association for the Advancement of Artificial Intelligence*, 2011.

[84] Richard E. Korf. Depth-first iterative-deepening: An optimal admissible tree search. *Artificial Intelligence 27 97-109*, 1985.

[85] I. Pohl. Bi-directional and heuristic search in path problems. May 1969.

[86] R. Munos. From bandits to Monte-Carlo tree search: The optimistic principle applied to optimization and planning. *INRIA Lille – Nord Europe*, Feb 2015.

[87] T. Schulte and T. Keller. Balancing exploration and exploitation in classical planning. *Proceedings of the Seventh Annual Symposium on Combinatorial Search*, 2014.

[88] G. Stoltz S. Bubeck, R. Munos and C. Szepesvari. X-armed bandits. *Journal of Machine Learning Research 12 (2011) 1655-1695*, 2011.

[89] Robert L.Schultz and Nelson R.Zagalsky. Aircraft performance optimization. *Journal of Aircraft*, 1972.

[90] ARINC. Navigation system database. *Aeronautical Radio Incorporated, ARINC Specification 424-22*, July 2018.

[91] Eurocontrol. *User Manual For the Base of Aircraft Data (BADA) Revision 3.9: EEC Technical/Scientific Report No. 11/03/03-03*. Eurocontrol Experimental Center (EEC), April 2011.

[92] R. Andreu-Altava, J.C. Mere, Daniel Delahaye, and Thierry Miquel. Graph-search descent and approach trajectory optimization based on enhanced aircraft energy management. *AIAA Aviation Forum*, June 2019.

[93] J. Pearl. *Intelligent Search Strategies for Computer Problem Solving*. Addison-Wesley Publishing Company, 1984.

[94] H. Antikainen. Using the hierarchical pathfinding A* algorithm in GIS to find paths through rasters with nonuniform traversal cost. *ISPRS Int. J. Geo-Inf.*, 2013.

[95] Garret D. Fett. Aircraft route optimization using the A* algorithm. *Department of the Air Force air University*, March 2014.

[96] W. Maazoun. Conception et analyse d'un système d'optimisation de plans de vol pour les avions. *Département de mathématiques et de génie industriel École Polytechnique de Montréal*, April 2015.

[97] R. Howe-Veenstra and Y. J. Zhao. Commercial aircraft trajectory optimization and efficiency of air traffic control procedures.

[98] O. Stursberg. A graph search algorithm for optimal control of hybrid systems. *43rd IEEE Conference on Decision and Control*, December 14-17, 2004.

[99] R. L. Rivest T. H. Cormen, C. E. Leiserson and C. Stein. *Introduction to Algorithms*. The MIT Press. Cambridge Massachusetts., 2009.

[100] R. Andreu-Altava, J.C. Mere, Daniel Delahaye, and Thierry Miquel. Flight management system pathfinding algorithm for automatic trajectory generation. *37th Digital Avionics System Conference, London, UK*, September 2018.

[101] Jarvis A., H.I. Reuter, A. Nelson, and E. Guevara. Hole-filled seamless srtm data v4, international centre for tropical agriculture (ciat). `http://srtm.csi.cgiar.org`, 2008.

[102] Yashovardhan S. Chati and Hamsa Balakrishnan. Analysis of aircraft fuel burn and emissions in the landing and take off cycle using operational data. *6th International Conference on Research in Air Transportation (ICRAT)*, 2014.

[103] M. Schaefer. Development of a forecast model for global air traffic emissions. *Deutsches Zentrum für Luft- und Raumfahrt Institut für Antriebstechnik Köln*, June 2012.

[104] D.Lee M. Mann et al. J. Faber, D. Greenwood. Lower nox at higher altitudes. policies to reduce the climate impact of aviation nox emission. *Deutsches Zentrum für Luft- und Raumfahrt Institut für Antriebstechnik Köln*, October 2008.

[105] J. R. Merkt. Flight energy management training: Promoting safety and efficiency. *Journal of Aviation Technology and Engineering 3:1 24–36*, 2013.

[106] ICAO. *State of Global Aviation Safety*. ICAO Safety Report. International Civil Aviation Organization, 999 Robert-Bourassa Boulevard, Montréal, Quebec, Canada H3C 5H7, 2019.

[107] EASA. *Annual Safety Review*. 2018.

[108] FSF. Stabilized approach. *Flight Safety Foundation ALAR Briefing Note 7.1*.

[109] Tzvetomir Blajev and Capt. William Curtis. *Go-Around Decision-Making and Execution Project*. Flight Safety Foundation, March 2017.

[110] J.C. Mere R. Andreu Altava. Procédé et dispositif pour générer en temps réel une trajectoire verticale optimale destinée à être suivie par un aéronef.

[111] ICAO. Standard atmosphere. *ISO 2533:1975*, 1975.

[112] Eurocontrol. *The Advanced Emission Model (AEMIII) Version 1.5*. Eurocontrol Experimental Center (EEC), 2004.

[113] Doug DuBois and Gerald C. Paynter. Fuel flow method2 for estimating aircraft emissions. *SAE Transactions. Journal of Aircraft. Vol.115, Section 1, pp. 1-14*, 2006.